

UNIVERSIDADE DE LISBOA



**A Robótica Educativa no Ensino e Aprendizagem de
Conceitos de Programação e Algoritmos**

Diana Filipa Rodrigues Oliveira

Relatório da Prática de Ensino Supervisionada

Mestrado em Ensino de Informática

2013

UNIVERSIDADE DE LISBOA



**A Robótica Educativa no Ensino e Aprendizagem de
Conceitos de Programação e Algoritmos**

Diana Filipa Rodrigues Oliveira

Relatório da Prática de Ensino Supervisionada
Orientada pelo Professor Doutor João Filipe Matos

Mestrado em Ensino de Informática

2013

*"Um professor afeta a eternidade...
é impossível dizer até onde vai sua influência."*

Henry Adams

Agradecimentos

As minhas primeiras palavras de agradecimento são dirigidas para a minha família. Obrigada por acreditarem em mim e pelo vosso apoio incondicional ao longo destes dois anos, com especial ênfase para o meu marido Tiago Silva e a minha irmã Dulcineia Fernandes por pacientemente lerem o meu relatório e ajudarem-me nas revisões. Em seguida tenho que agradecer às minhas colegas de curso e amigas para toda uma vida, Susana Ferreira e Honorina Celestino, com quem partilhei expetativas e ansiedades, e que estiveram presentes nas minhas vitórias e derrotas. Obrigada pela vossa amizade e partilha!

Agradeço ao professor João Filipe Matos pela sua orientação e abertura para aceitação deste estudo. De uma forma muito especial, agradeço à professora Paula Abrantes pela sua visão e capacidade de nos fazer querer ir mais além. Agradeço-lhe por toda a disponibilidade e pelas imensas reflexões conjuntas que muito contribuíram para a minha construção como professora.

Obrigado à Escola Secundária de Camões por ter permitido realizar a prática de ensino supervisionada nas suas instalações e em especial à professora Mónica Batista por amavelmente abrir portas na sua sala de aula. Por último e não menos importante aos alunos que fizeram parte desta experiência, esperando que tenha sido tão enriquecedor para eles quanto foi para mim.

Resumo

O presente relatório descreve a prática de ensino supervisionada realizada do âmbito do Mestrado de Ensino de Informática, numa turma de 12º ano do curso Científico Humanístico de Ciências e Tecnologias da Escola Secundária de Camões à disciplina Aplicações Informáticas B, na unidade de ensino-aprendizagem Introdução à Programação, mais especificamente aos conteúdos: programação e algoritmos.

Face as elevadas taxas de insucesso escolar em disciplinas introdutórias de programação, a intervenção letiva foi planeada utilizando a robótica educativa, seguindo uma metodologia de aprendizagem baseada por resolução de problemas envolto numa temática de super-heróis. Com o intuito de minimizar os problemas de ensino e aprendizagem de programação de utilidade, fazendo os alunos programar com a finalidade de salvar uma cidade; sintaxe, por utilizarem linguagens de programação visual; e preferência por aulas de componente prática, por promover o máximo de momentos práticos com o robô possíveis.

Ao longo de cinco aulas de noventa minutos os alunos resolveram dois desafios de complexidade crescente que serviam de base para a sua avaliação. As classificações obtidas foram cem por cento positivas e bastante satisfatórias. A análise de dados indica que os alunos gostam de programar o robô, gostam dos desafios e da maneira como foram propostos. O estudo sobre o impacto da robótica educativa segundo as perspetivas dos alunos, indicou que o robô ajudou-os a manter a sua atenção nas aulas, influenciando positivamente os seus resultados.

Palavras-chave: Programação, Algoritmos, Robótica Educativa, Lego Mindstorms.

Abstract

This report describes the supervised teaching practice in the context of the Master in Informatics Teaching, carried out in a class of 12th year of Science and Technology course in Escola Secundária de Camões School, at the discipline Computer Applications B, in the unity of teaching-learning Introduction to Programming, specifically the contents: programming and algorithms.

Given the high rates of school failure in introductory programming courses, the intervention was planned using educational robotics, and a methodology based in learning by problem solving wrapped in a theme of superheroes. In order to minimize the problems of teaching and learning programming: utility, making students program the robot with the purpose of saving a city; syntax, by using visual programming languages; and preference for practical component classes, allowing students to interact with the robot possible the most time possible.

Over five lessons of ninety minute students solved two challenges of increasing complexity that serves as a basis for their evaluation. The ratings obtained were a hundred percent positive and quite satisfactory. The data analysis indicates that students like to program the robot, enjoy the challenges and the way they were proposed. The study on the impact of educational robotics according to the perspectives of the students indicated that the robot helped them to keep their attention in class, positively influencing their results.

Keywords: Programming, Algorithms, Educational Robotics, Lego Mindstorms.

Índice

1. Introdução	1
2. Contexto da Prática de Ensino Supervisionada.....	4
2.1. Caracterização da Escola	5
2.2. Caracterização do Curso	6
2.3. Caracterização da Disciplina	8
2.4. Caracterização da Unidade de Ensino-Aprendizagem	9
2.5. Caracterização da Turma	10
3. Enquadramento curricular e didático da prática de ensino supervisionada	13
3.1. A importância do pensamento computacional.....	14
3.2. Definição de programação	15
3.3. Conceitos de programação e suas vantagens	16
3.4. Definição de algoritmo	16
3.5. Problemática no Ensino-Aprendizagem de Programação.....	18
3.6. Robótica educativa.....	19
3.7. Linguagens de programação visual.....	21
3.8. Robô Lego® Mindstorms® NXT	22
3.8.1 Programação no i-Brick e NXT-G.....	23
3.8.2 Bloco MOVE.....	25
3.9. Contextos de aprendizagem	26
3.10 Aprendizagem por resolução de problemas e storytelling.....	27
4. Planeamento da Prática de Ensino Supervisionada.....	28
4.1. Temática geral e específica dos desafios	29
4.2. Plano geral da prática de ensino supervisionada.....	33
4.3. Objetivos de aprendizagem.....	35
4.4. Plano de Avaliação	38
4.6. Estratégias de Ensino	42
4.5. Método de trabalho	45
5. Narrativas da Prática de Ensino Supervisionada.....	47
5.1. Concretização das aulas	48
5.1.1. Narrativa da primeira aula.....	48
5.1.2. Narrativa da segunda aula.....	52
5.1.3. Narrativa da terceira aula.....	56

5.1.4.	Narrativa da quarta aula.....	59
5.1.5.	Narrativa da quinta aula.....	60
5.2.	Avaliação da Prática de Ensino Supervisionada.....	61
5.2.1.	Avaliação do Desafio 1.	61
5.2.2.	Avaliação do Desafio 2.	64
6.	Cariz investigativo da prática de ensino supervisionada.....	66
6.1	Problemática da dimensão investigativa	66
6.2	Abordagem metodológica	67
6.2.1.	Caraterização da amostra.....	68
6.2.2.	Variáveis em estudo.	69
6.2.3.	Metodologias de recolha e análise de dados.....	69
6.2.4.	Descrição do instrumento e processo de recolha de dados.....	71
6.3	Apresentação e análise de dados.....	73
6.3.1	Dimensão atitudes e perceções.....	73
6.3.2	Dimensão atenção.....	76
6.3.3	Questões abertas.	77
6.4	Discussão e considerações	78
7.	Reflexão da Prática de Ensino Supervisionada.....	80
	Referências	83
	Anexos.....	i
	Anexo A: Questionário de caraterização da turma.....	ii
	Anexo B: Autorizações para a realização da prática de ensino supervisionada.....	vii
	Anexo C: Formulário de Observação de Aulas do Professor Cooperante	xii
	Anexo D: Kit da Lego Mindstorms NXT	xiv
	Anexo E: Planificação da prática de ensino supervisionada	xviii
	Anexo F: Taxonomia dos objetivos de aprendizagem.....	xxv
	Anexo G: Critérios de avaliação da disciplina Aplicações Informáticas B.....	xxx
	Anexo H: Grelhas das Atitudes e comportamentos.....	xxxii
	Anexo I: Regras de avaliação	xxxiv
	Anexo J: Exemplo de aplicação das regras de avaliação.....	xxxviii
	Anexo K: Disciplina NXT HEROES - Moodle	xl
	Anexo L: Apresentação sobre conceitos de programação - preenchido.....	xlii
	Anexo M: Apresentação sobre conceitos de algoritmos - preenchido	xliv

Anexo N: Exemplo de Folha de Reflexão de algoritmos	xlvi
Anexo O: Modelos de robôs da intervenção letiva	xlvi
Anexo P: Desafio NXT HEROES	l
Anexo Q: Envelopes	lv
Anexo R: Folha de Testes do i-Brick,	lvii
comparação entre programação no i-Brick e NXT-G	lvii
Anexo S: Pré Questionário Atitudes e Percepções	lx
Anexo T: Exemplo de feedback escrito	lxii
Anexo U: Questionário Desafio 1 e análise	lxiv
Anexo V: Questionário Desafio 2 e análise	lxix
Anexo W: Pós Questionário sobre atitudes e percepções e atenção	lxxiii
Anexo X: Grelhas de avaliação da prática de ensino supervisionada	lxxvi
Anexo Y: Análise das questões abertas do pós questionário	lxxx
Anexo Z: Complementos da narrativa da pratica de ensino supervisionada	lxxxiv

Índice de Quadros

Quadro 1. <i>População escolar da Escola Secundária de Camões</i>	5
Quadro 2. <i>Matriz curricular do 12º ano de Curso Científico-Humanístico Ciências e Tecnologias da Escola Secundária de Camões</i>	7
Quadro 3. <i>Preferências dos alunos em relação à disciplina, unidade de ensino-aprendizagem e robôs</i>	12
Quadro 4. <i>Síntese do plano a curto prazo da prática de ensino supervisionada</i>	34
Quadro 5. <i>Plano de avaliação final da prática de ensino supervisionada</i>	39
Quadro 6. <i>Plano de avaliação do Desafio 1</i>	40
Quadro 7. <i>Plano de avaliação da atividade extra</i>	41
Quadro 8. <i>Plano de avaliação do Desafio 2</i>	41
Quadro 9. <i>Listagem de recursos e materiais para a prática de ensino supervisionada</i>	41
Quadro 10. <i>Classificações finais do Desafio 1</i>	62
Quadro 11. <i>Médias de classificações por parâmetro do Desafio 1</i>	62
Quadro 12. <i>Classificações do parâmetro Reflexão do Desafio 1</i>	63
Quadro 13. <i>Classificações finais do Desafio 2</i>	64
Quadro 14. <i>Comparação de médias de classificações por parâmetro</i>	65
Quadro 15. <i>Dimensão atitudes e perceções</i>	74
Quadro 16. <i>Dimensão atenção</i>	76

Índice de Figuras

Figura 1. <i>Escola Secundária de Camões</i>	5
Figura 2. <i>Distribuição de disciplinas que os alunos mais gostam e menos gostam</i>	10
Figura 3. <i>Distribuição da área de preferência</i>	11
Figura 4. <i>Distribuição por motivos de escolha da disciplina de Aplicações Informáticas B</i>	11
Figura 5. <i>Fases do Pedagogical Content Knowledge</i>	13
Figura 6. <i>Motores, sensores e i-Brick do kit Lego Mindstorms NXT</i>	22
Figura 7. <i>Programação direta no i-Brick</i>	23
Figura 8. <i>Interface gráfica do NXT-G</i>	24
Figura 9. <i>Propriedade duração do bloco MOVE</i>	25
Figura 10. <i>Modelo de Evento de Aprendizagem adaptado</i>	26
Figura 11. <i>Capítulo 1 da história NXT-Heroes</i>	30
Figura 12. <i>Capítulo 2 da história NXT Heroes</i>	30
Figura 13. <i>Capítulo 3 da história NXT Heroes</i>	30
Figura 14. <i>Proposta Desafio 1</i>	31
Figura 15. <i>Proposta Desafio 2</i>	32
Figura 16. <i>Curva de performance de equipas</i>	46
Figura 17. <i>Cenário do Desafio 2</i>	56
Figura 18. <i>Exemplificação de uma rotação no robô em graus</i>	58
Figura 19. <i>Robô basquetebolista alterado</i>	59
Figura 20. <i>Análise de classificações entre Definir programação e Aplicação de Conceitos de programação</i>	64
Figura 21. <i>Correlação entre programar e programar com robôs</i>	75

1. Introdução

Independente do grau e do sistema de ensino, as disciplinas onde são ensinados conceitos básicos de programação têm sido assombradas por elevadas taxas de insucesso escolar (Gomes, Henriques & Mendes, 2008). Do ponto de vista dos professores o ato de ensinar programação por si só não é fácil, uma vez que devido a complexidades individuais, diferentes alunos revelam diferentes dificuldades a nível da programação (Ala-Mutka, 2003). Entre as quais a dificuldade em compreender a utilidade da programação e aplicar corretamente a sintaxe das linguagens de programação (Gomes et al., 2008; Saeli, Perrenet, Jochems & Zwaneveld, 2011). Por outro lado, os alunos manifestam preferência por aulas com maior componente prática, em detrimento de uma exposição de conteúdos excessiva (Gomes, Martinho, Bernardo, Matos, & Abrantes, 2012). Estes fatores induzem muitos a considerar a programação como difícil de aprender.

Ensinar para Roldão (2009) envolve “desenvolver uma ação especializada, fundada em conhecimento próprio, de fazer com que alguém aprenda alguma coisa que se pretende e se considera necessária” (p. 14). Saeli e colaboradores (2011) afirmam que é fundamental no ensino de programação o professor ser detentor de conhecimentos científicos de programação e encontrar a utilidade prática neles. Depois ao centrar-se num conteúdo específico e face às dificuldades de aprendizagem procurar formas de colmatá-las, organizando um conjunto variado de dispositivos que possam promover a aprendizagem (Roldão, 2009; Saeli et al., 2011). Desta forma com um ensino suportado por ferramentas e estratégias de ensino adequadas, Papert (1980) defende que é possível em certa medida promover alunos peritos em programação. O professor ao tornar pública a sua experiência de ensino e aprendizagem passa a fazer parte do ciclo de investigação da Didática da Informática, mais especificamente da programação.

De entre as escolas com protocolo com o Mestrado de Ensino de Informática da Universidade de Lisboa, para a realização da prática de ensino supervisionada, encontrava-se a Escola Secundária de Camões [ESC], que face à extinção da disciplina Área de Projeto conhecida pelo seu sucesso na implementação da robótica educativa, mostrou interesse em reavivar a investigação da programação com auxílio da robótica na disciplina de 12º ano de Aplicações Informáticas B, do curso Científico-Humanístico de Ciências e Tecnologias (ESC, 2010; ESC, 2012). Assim, da junção do meu interesse pessoal pela didática da programação, com o historial de robótica educativa aliada à

programação da Escola Camões nasceu esta prática de ensino supervisionada, que decorreu ao longo de cinco aulas de noventa minutos na unidade de ensino-aprendizagem Introdução à Programação, que tem duração total de trinta horas e os conteúdos programáticos abordados foram conceitos de programação e algoritmos (Pinto, Dias & João, 2009).

Estudos realizados pela Associação Japão Robótica [JPA], a Comissão Económica das Nações Unidas [UNEC] e da Federação Internacional de Robótica [IFR], indicam que o crescimento do mercado dos robôs pessoais, incluindo aqueles usados para fins educativos e de entretenimento, tem sido enorme e prevê-se que a tendência continue nas próximas décadas (Kara, 2004, citado por Benitti, 2012). A revisão de literatura revela que no ensino básico a robótica tem sido utilizada como ferramenta para o ensino e aprendizagem de outras disciplinas escolares, com especial ênfase em projetos interdisciplinares envolvendo Ciência, Matemática, Informática e Tecnologia. Enquanto no ensino secundário e superior a robótica é mais utilizada como objeto de aprendizagem, incidindo sobre disciplinas de programação, inteligência artificial e mecatrónica (Mitnik, Nussbaum & Soto, 2008, citado por Benitti, 2012). Relatos de professores que utilizaram a robótica educativa indicam que os robôs fascinam os alunos e que esse interesse pode ser usado como um incentivo para a aprendizagem, visto os alunos quererem ver o sucesso da sua invenção (Kumar & Meeden, 1998) e que para além disso, os robôs incentivam os alunos a aprenderem conceitos fora do currículo (Klassner, 2002, citado por Koski et al., 2008). Vários são os que defendem que atividades de robótica tem um enorme potencial para melhorar o ensino em sala de aula, criando melhores currículos e currículos mais inspiradores (Lego, 2006; Papert, 1993). Rusk e colaboradores (2008) estudaram estratégias para introduzir os alunos na robótica e defendem a importância de proporcionar contextos de aprendizagem de múltiplas soluções que vão de encontro com os diversos interesses dos alunos, de modo envolvê-los.

Tendo como finalidade explorar as potencialidades da utilização da robótica educativa à disciplina de Aplicações Informática B, no ensino e aprendizagem de conceitos de programação e algoritmos, foi projetada uma planificação utilizando como contexto de aprendizagem a aprendizagem baseada por problemas e o *storytelling*, tendo como tema uma história de super-heróis denominada NXT Heroes, onde os alunos tentam salvar a sua cidade que está em perigo, pela personificação e programação de heróis-robóticos. Assim, iniciam uma “Semana de Treino” constituída por dois desafios, onde os

alunos participam na construção dos problemas base e depois criam as suas próprias soluções, com o objetivo de cumprir a sua missão. Finda a prática de ensino supervisionada, realizou-se uma análise sobre o impacto da robótica educativa no ensino e aprendizagem de programação do ponto de vista dos alunos, medido através das variáveis atitudes, percepções e atenção.

A organização do relatório da prática de ensino supervisionada encontra-se repartida por sete capítulos, iniciando-se com a presente Introdução. O capítulo dois apresenta uma caracterização da escola, curso, disciplina, unidade de ensino-aprendizagem e turma. No capítulo três são analisados os conhecimentos científicos ligados à subunidade lecionada e é apresentada a problemática de ensino de programação e as soluções propostas de modo a promover o ensino e aprendizagem. O capítulo quatro descreve o plano de intervenção, fundamentando-o através da apresentação dos objetivos de aprendizagem, estratégias pedagógicas, recursos e instrumentos de avaliação. O capítulo cinco apresenta uma descrição sumária das aulas realizadas em forma de narrativa e a sua relação com as aprendizagens dos alunos através da avaliação. No capítulo seis é explorada a dimensão de cariz investigativo descrevendo os procedimentos e os instrumentos de recolha e análise de dados, sendo no final apresentados e discutidos os dados recolhidos no processo. Por último, no capítulo sete apresenta-se uma reflexão sobre o processo de escrita da intervenção letiva e a sua concretização. No final do relatório podem ser consultadas as referências bibliográficas, assim como os anexos que compõem e dão evidência de todo o trabalho desenvolvido.

2. Contexto da Prática de Ensino Supervisionada

Nos últimos anos as escolas têm passado por um reforço na autonomia, liberdade de organização e diversidade curricular, de modo a responderem às necessidades dos alunos e das comunidades locais (Nóvoa, 2006). Em consequência, hoje não existe um conceito único de escola, tal como referência Nóvoa (2006) “*Não há uma escola, há muitas escolas*”. E consequentemente não existe um único método de ensino “*Não há uma solução, há muitas soluções*”. Assim, este relatório inicia-se com uma análise dos elementos que caracterizam a escola onde decorreu a prática de ensino supervisionada, que permitiram planear um método de ensino e aprendizagem que coaduna-se com o lema da escola, as exigências do curso e da disciplina, os interesses dos alunos e principalmente que promovesse harmonia entre a intervenção letiva e as aulas da professora cooperante.

A intervenção letiva decorreu na Escola Secundária de Camões, no 12º ano do curso Científico-Humanístico de Ciência e Tecnologia, na disciplina de Aplicações Informáticas B. A recolha de dados para toda a caracterização foi elaborada através da análise dos instrumentos de autonomia escolar disponibilizados pela escola, e outros documentos organizacionais relevantes, tais como, os dossiês de turma. De modo a obter dados suplementares aos encontrados nos documentos enunciados, foi elaborado um questionário de caracterização da turma (Anexo A), que foi posteriormente preenchido *online* por todos os alunos da turma. Com fim de salvaguardar a confidencialidade dos dados dos alunos e obter um consentimento informado¹ para a recolha de dados foi formalizado um pedido de autorização aos encarregados de educação, igualmente com autorização do diretor da escola, e com o conhecimento dos diretores de turma e da professora cooperante (Anexo B). Para além disso, foi elaborado um registo de observação de aulas da professora cooperante (Anexo C) com base nas grelhas de observação de aulas e avaliação docente desenvolvidas por Pedro Reis (2011), onde através da observação não participativa² foi possível enriquecer os dados referentes à turma em questão, no que diz respeito a padrões de comportamento na sala de aula.

¹ O consentimento informado envolve, segundo Tuckman (2005) a compreensão dos procedimentos, a compreensão dos riscos e a compreensão dos direitos dos sujeitos; por parte dos intervenientes, ou dos seus encarregados de educação quando menores.

² Segundo Mack, Woodson, Macqueen, Guest, & Namey (2005) observação não participativa é quando o investigador não interage com o objeto em estudo.

2.1. Caraterização da Escola

Enobrecida pelos seus mais de cem anos de história, a Escola Secundária de Camões é uma das mais antigas e prestigiadas escolas de Lisboa, criada em 1902 por Carta de Lei de 24 de Maio, inicialmente com o nome de Liceu Nacional de Lisboa (ESC, 2010). Várias reformas de caráter organizacional e estrutural decorreram com objetivo de prover um ensino de melhor qualidade, intenção denotada pelo lema do Projeto Educativo em rigor “Para espertar engenhos curiosos” (ESC, 2010, p.1).

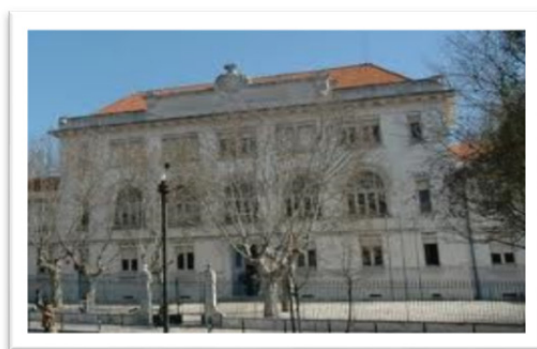


Figura 1. *Escola Secundária de Camões*

Hoje, a escola encontra-se localizada na Freguesia de São Jorge de Arroios integrada numa zona residencial e de serviços (ESC, 2010) (Figura 1). A sua diversidade de população escolar deve-se aos seus acessos (ESC, 2010). No início do 1º Período do ano letivo 2012/2013 a escola funcionava com 142 professores, 13 assistentes técnicos e 34 assistentes operacionais (ESC, 2013) e estavam matriculados 1866 alunos, sendo 1130 de cursos Científico-Humanísticos Tecnológico de Desporto e Ensino Profissional, 491 de Ensino Recorrente por Módulos, CEF e EFA Escolar e 245 Formações Modulares, Português para Todos (ESC, 2013) (Quadro 1).

Quadro 1. *População escolar da Escola Secundária de Camões*

Modalidade de ensino	2009-2010	2010-2011	2011-2012	2012-2013
Científico-Humanísticos, Tecnológico de Desporto, Ensino Profissional	1073	1039	1162	1130
Ensino Recorrente por Módulos, EFA básico e secundário.	357	417	301	491
Formações Modulares, Português para Todos	225	277	260	245
TOTAL	1655	1733	1723	1866

A nível de oferta curricular no ano letivo 2012/2013, a escola conta com 43 turmas no ensino diurno, nos seguintes cursos: 16 turmas de Ciências e Tecnologias; 6 turma de Ciências Socioeconómicas; 3 turmas de Artes Visuais; 9 turmas de Línguas e Humanidades; 1 turma do Curso Tecnológico de Desporto; 2 turmas do Curso Profissional Técnico de Gestão Desportiva; 3 turmas do Curso Profissional Técnico de Serviços Jurídicos e 3 turmas do Curso Profissional de Informática de Gestão. No ensino noturno, encontram-se em funcionamento os seguintes cursos: 10 turmas de Ensino Recorrente; 1 turma de Cursos EFA Básico e 4 turmas de EFA Secundário; 5 turmas de Formações Modulares de Línguas e 4 turmas de Português para Todos (ESC, 2013).

De entre estes cursos, os disponíveis para realização da prática de ensino supervisionada constavam o Curso de Ciências de Tecnologias à disciplina de 12º ano Aplicações informáticas B ou a de 11º ano Sistemas de Informação. Estando também disponível o Curso Profissional de Informática de Gestão a disciplina de 10º ano linguagens de programação. Em acordo com o professor orientador e professora cooperante, a escolha recaiu sobre a disciplina de Aplicações Informáticas B, que entrou em funcionamento pela primeira vez no presente ano letivo e que almejava a implementação da robótica educativa na unidade de ensino-aprendizagem de introdução à programação (ESC, 2012).

2.2. Caraterização do Curso

O ensino secundário tem como finalidade assegurar uma educação comum e, ao mesmo tempo uma formação especializada numa direção, com diferentes implicações em termos de prosseguimento de estudos e de oportunidades de realização profissional (Ribeiro & Ribeiro, 1990). Mais especificamente, os Cursos Científico-Humanísticos regulados pela Portaria n.º 550-D/2004, de 21 de Maio, com as alterações introduzidas pela Portaria n.º 259/2006, de 14 de Março, têm como objetivo dotar os alunos de um conjunto de saberes e competências que lhes possibilitem observar e organizar o espaço envolvente, bem como compreender a ciência e a tecnologia reconhecendo e relacionando as suas implicações na sociedade contemporânea. Vocacionados para o prosseguimento de estudos de nível superior os cursos Científico-Humanísticos têm a duração de 3 anos letivos correspondentes aos 10.º, 11.º e 12.º anos de escolaridade (Ministério da Educação e Ciência, 2011a).

A matriz curricular dos Cursos Científico-Humanísticos integra duas componentes principais: a de formação geral e a de formação específica. Quanto à componente de formação geral, esta é comum a todos os cursos, e visa a construção da identidade pessoal, social e cultural dos jovens e inclui duas disciplinas trienais: Português e Educação Física (10º, 11º e 12º ano), e duas bienais: Língua Estrangeira I, II ou III e Filosofia (10º e 11º ano). Enquanto a componente de formação específica, visa proporcionar formação científica consistente no domínio do respetivo curso, e integra para além de uma disciplina trienal (10º, 11º e 12º ano), duas disciplinas opcionais bienais (10º e 11º ano) e duas disciplinas opcionais anuais (12º ano), sendo a escolha e combinação das disciplinas opcionais da responsabilidade do aluno, em função do percurso formativo que pretende (Ministério da Educação e Ciência, 2011a; 2011b).

Quadro 2. Matriz curricular do 12º ano de Curso Científico-Humanístico Ciências e Tecnologias da Escola Secundária de Camões

Componentes de Formação	Carga horaria semanal organizada em períodos de 45 minutos para o 12º ano
Geral:	
Português	5
Educação Física	4
Específica:	
Trienal	6 (Matemática)
Opção Anual 1	4 (Biologia, Física, Química ou Geologia)
Opção Anual 2	4 (Anual 1, Psicologia B ou Aplicações Informáticas B)

Mais especificamente no 12º ano do Curso Científico-Humanístico de Ciências e Tecnologias (Quadro 2), os alunos terão as disciplinas trienais Português e Educação Física de componente geral, juntamente com a disciplina trienal Matemática A e mais duas disciplinas anuais opcionais de componente específica (Ministério da Educação e Ciência, 2011a; 2011b). Referente às disciplinas opcionais, a primeira obrigatoriamente terá ser uma entre: Biologia, Física, Química e Geologia; a segunda pode ser outra das anteriores, ou uma oferecida pela escola, estando neste caso disponíveis: Psicologia B e Aplicações Informáticas B (Ministério da Educação e Ciência, 2011a; 2011b). Este leque variado de disciplinas faz com que o curso tenha várias vertentes de saídas profissionais, desde a área das tecnologias, engenharias, saúde e ciências.

2.3. Caraterização da Disciplina

Apesar da literacia digital continuar a sua ascensão como competência fundamental em cada área disciplinar e em cada profissão, acreditar que os alunos estão a aprender informática porque existem computadores nas escolas, é como acreditar que eles estejam a aprender química, porque existem tubos de ensaio nos armários da escola (CSTA, 2012, Novembro; Pinto et al., 2009). Assim, a Computer Science Teacher Association [CSTA] (2012, Novembro) defende que são importantes disciplinas de componente informática ao longo do percurso escolar dos alunos, uma vez que só porque os jovens usam tecnologia, não significa que a entendem ou são capazes de criá-la.

Em Portugal, uma das disciplinas de componente informática existentes é a disciplina de Aplicações Informáticas B homologada por despacho do Senhor Secretário de Estado da Educação, pelo Decreto-Lei n.º 272/2007, de 26 de Julho como disciplina anual de opção do 12º ano dos cursos Científico-humanísticos de Ciências e Tecnologias, Ciências Socioeconómicas, Línguas e Humanidades e Artes Visuais. Esta viu reduzida a sua carga horária semanal três de para dois blocos de 90 minutos de acordo com o Decreto-Lei n.º 129/2012, de 5 de Julho (Pinto et al., 2006). Em Julho de 2009 foi homologado o programa de Aplicações Informáticas B em vigor, composto por quatro unidades de ensino-aprendizagem fixas: Introdução à Programação, Introdução à Teoria da Interatividade, Conceitos Básicos Multimédia e Utilização dos Sistemas Multimédia (Pinto et al., 2009). Visto os alunos que optem por esta disciplina serem de áreas do saber distintas, o programa recomenda ao professor adequar o grau de amplitude de desenvolvimento dos conteúdos programáticos conforme o público que possui na sala de aula, em articulação com o Conselho de Turma (Pinto et al., 2009).

A Escola Secundária de Camões considerou um erro a extinção da disciplina Área de Projeto pelo Decreto-Lei n.º 50/2011, na medida em esta era um “Espaço de Paixões” e “descobertas”, assim sendo, como forma de rentabilizar os recursos de robôs existentes e de continuar a investigação iniciada com a utilização da robótica no ensino e aprendizagem, o Grupo de Informática da escola propôs a abertura da disciplina Aplicações Informáticas B para o ano letivo 2012/2013 (ESC, 2010, 2012). Segundo a Proposta de oferta da disciplina de Aplicações Informáticas B para o ano letivo 2012/2013 (ESC, 2012) da Escola Secundária de Camões, a disciplina tem como objetivos principais: aprender os conceitos da programação com recurso ao robô Mindstorms NXT da Lego; reconhecer e identificar ambientes de realidade virtual e a

forma como se relacionam com a interatividade; e identificar os diferentes sistemas multimédia e as suas características e saber desenvolver uma solução multimédia.

2.4. Caraterização da Unidade de Ensino-Aprendizagem

Das quatro unidades de ensino-aprendizagem disponíveis para a disciplina de Aplicações Informáticas B, o momento de intervenção letiva decorreu no início da unidade de ensino-aprendizagem Introdução à Programação, composta por 20 unidades letivas de 90 minutos, somando um total de 30 horas (Pinto et al., 2009). Neste caso, correspondente ao início do segundo período do ano letivo de 2012/2013, tendo como objetivo geral fazer aprender conceitos da programação com recurso ao Robô Mindstorms® NXT da Lego®, com especial foco na aprendizagem de conceitos de programação e conceitos de algoritmos.

A CSTA (2012, Novembro) afirma que educar os alunos para os trabalhos do passado é uma receita para o desastre, tanto económico como social. A mesma associação (CSTA, 2012, Novembro) considera o ensino de programação importante uma vez que nos Estados Unidos existe um elevado grau de empregabilidade na área das Ciências da Computação e Matemática, com especial ênfase para os desenvolvedores de *software* e programadores. Na perspetiva de Targetjobs (2007, citado por Carvalho, Lima & Fernandes, 2008) as competências mais valorizadas pelos empregadores são: capacidade de comunicação, capacidade de trabalhar em equipa, capacidade de gerir e liderar equipas, capacidade para resolver problemas, espírito de iniciativa e consciência comercial. Algumas destas competências são estimuladas na disciplina de Aplicações Informática B, em especial reforçadas pelo ensino de programação, uma vez que compõe as principais orientações didáticas do programa da disciplina, que assentam fortemente num carácter de cooperação em grupo, na realização de tarefas e na pesquisa de soluções para situações concretas ou problema propostos, sustentada em exemplos de carácter predominantemente prático e experimental (Pinto et al., 2009). Onde se pretende que os alunos possam, por um lado aprender os conceitos básicos de linguagem de programação, e por outro lado, possam aprofundar os seus conhecimentos numa linguagem de programação específica (Pinto et al., 2009).

2.5. Caraterização da Turma

Os alunos participantes na intervenção são provenientes de três turmas de 12º ano do Cursos Científico-Humanístico de Ciências e Tecnologias, correspondentes àqueles que escolheram a disciplina de opção Aplicações Informáticas B. No presente ano letivo a turma é composta por 22 alunos, 10 da turma A, 4 da turma B e 8 da turma E; com idades compreendidas entre os 17 e 19 anos, existindo apenas 4 raparigas do total. Todos os alunos são de nacionalidade Portuguesa à exceção de um é que Romeno, para além disso não existe nenhum aluno que usufrua de estatuto trabalhador estudante, nem que apresente necessidades educativas especiais. Todos os alunos afirmam ter um computador em casa com acesso à internet. Em relação ao número de retenções existem 5 alunos que ficaram retidos uma vez no ensino secundário.

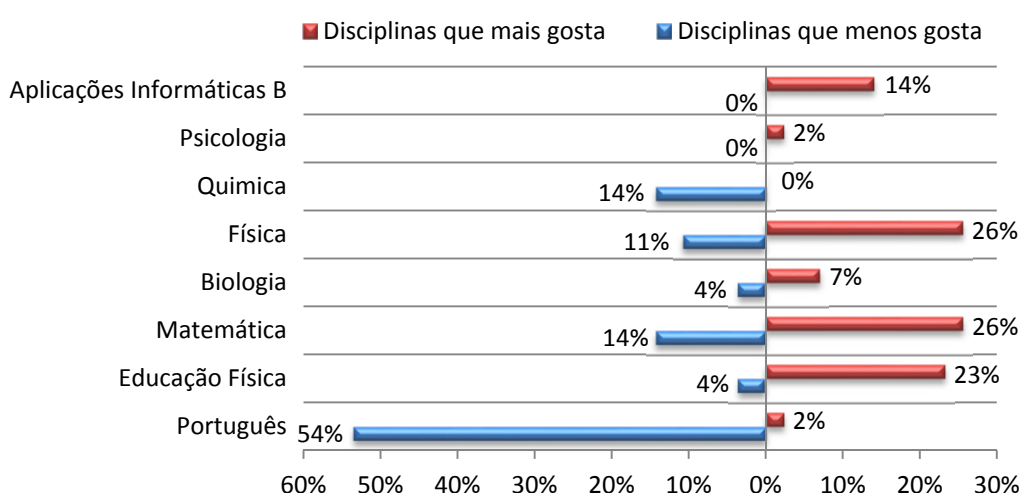


Figura 2. Distribuição de disciplinas que os alunos mais gostam e menos gostam

Referente às disciplinas do presente ano letivo, foi perguntado aos alunos através de questionário de caraterização da turma (Anexo A) quais as disciplinas de que gostavam mais e menos. Os resultados traduzidos no gráfico da Figura 2 indicam que as disciplinas que os alunos estavam a gostar mais eram Matemática e Física com igual percentagem de 26%. Por outro lado, a disciplina que os alunos estavam a gostar menos era maioritariamente Português com 54%. Mais especificamente em relação à disciplina de Aplicações Informáticas B, esta revelou não ser uma disciplina que os alunos depreciassem com 0%, como alias 14% dos alunos afirmavam ser a disciplina de que gostavam mais.

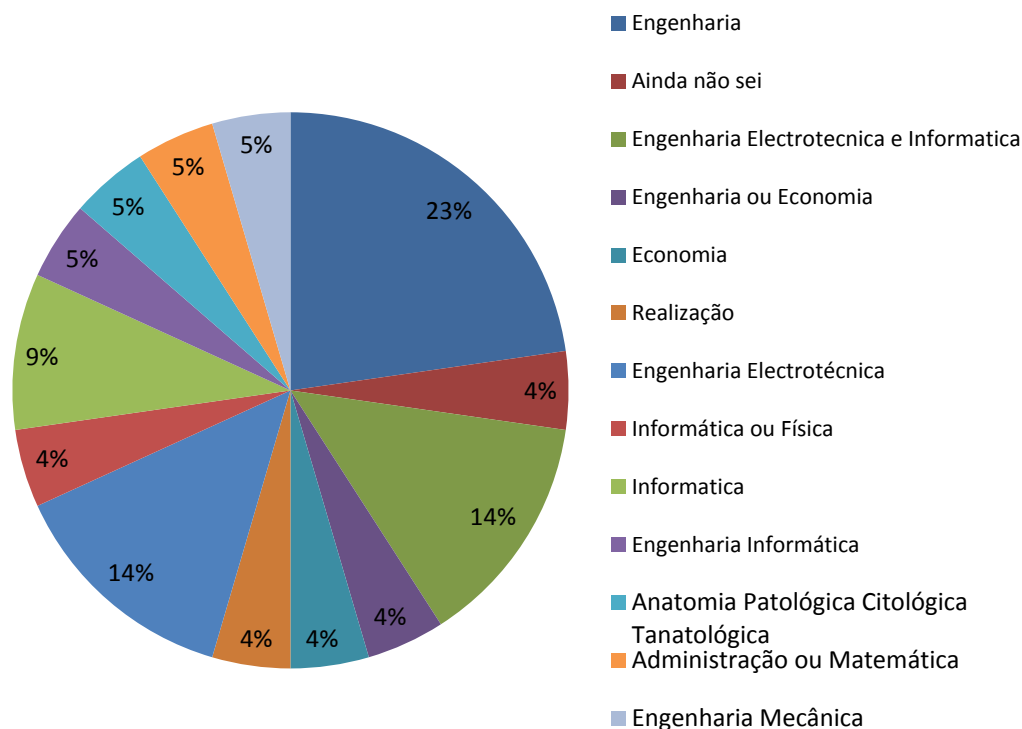


Figura 3. Distribuição da área de preferência

A importância do prosseguimento de estudos para assegurar o futuro profissional estava presente nestes alunos, visto todos afirmaram pretender continuar os estudos academicamente. Quando questionados sobre em que áreas pretendiam fazê-lo, apenas 4% afirmara ainda não saber, salientando-se que um total de 27%, o correspondente a 14% Engenharia Electrotécnica e Informática, com 9% Informática e 4% Engenharia Informática, alunos pretendiam prosseguir estudos numa área diretamente relacionada com informática (Figura 3).

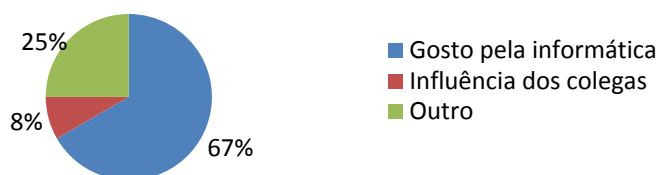


Figura 4. Distribuição por motivos de escolha da disciplina de Aplicações Informáticas B

Relativamente ao motivo que levou os alunos a escolher a opção de Aplicações Informáticas B, 67% dos alunos afirmaram ser pelo gosto pela informática, enquanto 8% afirmaram ter sido por influência de outros alunos e 25% e como 67% afirmaram ser por outros motivos: curiosidade, bases para o futuro académico, ser mais fácil e ter boas notas (Figura 4).

Em relação a se já haviam programado, dois alunos afirmam que sim em contexto extracurricular, com linguagens de programação, tais como: C, C++, Visual Basic, Java e Pawn. Inclusivamente, um desses alunos afirmou possuir um kit da Lego® Mindstorms® NXT em casa, apesar de que ainda não havia programado com ele. Segundo o ponto de visto dos alunos, a disciplina de Aplicações Informáticas B tem alguma importância, tal como, a unidade de ensino-aprendizagem Introdução à Programação. Para além disso, os alunos a afirmaram ter algum interesse na aprendizagem de programação com robôs (Quadro 3).

Quadro 3. Preferências dos alunos em relação à disciplina, unidade de ensino-aprendizagem e robôs

	Média	Desvio Padrão
Qual o grau de importância que o aluno atribui à disciplina de Aplicações Informáticas B?	2,95	0,49
Qual o grau de importância que o aluno atribui à unidade de ensino-aprendizagem Introdução à Programação?	3,18	0,59
Qual o grau de interesse do aluno em relação à utilização de robôs na unidade de ensino-aprendizagem Introdução à Programação?	3,32	0,78

Nota. Escala de quatro níveis

Com base formulário de registo de observações das aulas (Anexo C), foram analisadas quatro dimensões: organização da sala de aula, gestão da sala de aula, discurso do professor e discurso dos alunos. A nível de organização da sala de aula, a sala possuía espaço suficiente para comportar os 22 alunos e permitia uma disposição de mesas e cadeiras flexível. Os alunos sentavam-se em lugares fixos e estavam relativamente perto uns dos outros, e igualmente perto do professor. A sala estava equipada com computador, vídeo projetor, quadro branco, internet e robôs. A nível de gestão da sala de aula, os alunos aparentavam estar familiarizados com as regras de funcionamento da sala de aula, evidenciado pela facto de durante o decorrer das aulas os alunos não deslocarem dos seus lugares. Estavam organizados em grupos de dois elementos e aparentavam estar adaptados à organização de trabalho utilizada, por se agruparem ao elemento do grupo sem necessidade de aviso prévio. A nível do discurso do professor, durante as aulas as interações aluno-professor eram essencialmente em diálogo de grupo, onde aparentava ser proporcionado tempo de reflexão aos alunos. O discurso dos alunos por sua vez, às vezes direccionava-se para outros assuntos não de cariz educacional. A frequência com que os alunos faziam perguntas era média e todos pareciam receber a mesma atenção do professor. Para além disso, foi notado que os alunos utilizavam a plataforma Moodle.

3. Enquadramento curricular e didático da prática de ensino supervisionada

O “Pedagogical Content Knowledge” [PCK] é um conceito introduzido por Shulman (1986, 1987, citado de Saeli et al., 2011), que combina o conhecimento científico de um conteúdo, com o conhecimento pedagógico de como o ensinar, visando atingir melhores práticas educacionais. Para Roldão (2009, p.46) ensinar envolve “fazer aprender alguma coisa a alguém”, de facto um bom professor é aquele que tem um bom PCK, conseguindo transformar o seu conhecimento sobre determinado assunto em algo acessível para os alunos (Saeli et al., 2011). O PCK é um conhecimento que se desenvolve com anos de experiência de ensino (Saeli et al., 2011). Grossman (1989, 1990, citado por Saeli et al., 2011) divide o PCK em quatro fases distintas em forma de questões: Por que ensinar um determinado assunto? O que deve ser ensinado? Quais são as dificuldades de aprendizagem? Como ensinar? (Figura 5). As quatro questões estão ligadas entre si, visto que as razões para ensinar um tópico, influenciam o conteúdo escolhido a ser incluído no currículo (Saeli et al., 2011). Além disso, as dificuldades de aprendizagem dos alunos certamente influenciam a forma de ensiná-lo (Saeli et al., 2011).

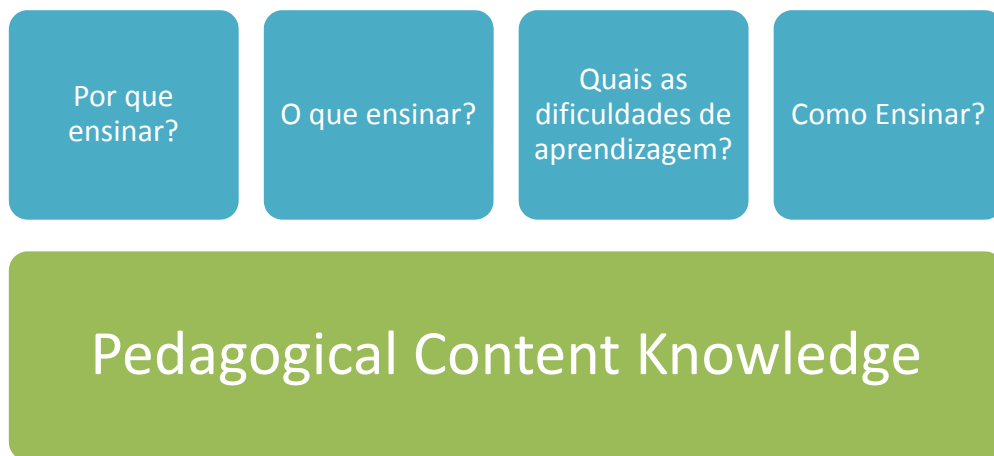


Figura 5. Fases do Pedagogical Content Knowledge

Este capítulo foi estruturado segundo o conceito de PCK começando por abordar a importância do ensino de conceitos de programação e algoritmos, passando a esmiuçar os conteúdos mais relevantes sobre conceitos de programação e algoritmos. Depois reflete sobre as dificuldades e barreiras no ensino e aprendizagem de programação do ponto de vista dos professores e alunos. Finalizando com uma reflexão sobre a sua didática e pedagogia fundamentada sobre formas de contornar essas dificuldades.

3.1. A importância do pensamento computacional

Uma das vantagens do ensino de programação e algoritmos é promover o pensamento computacional. O pensamento computacional combina a capacidade de pensamento humano com as capacidades dos computadores (Phillips, 2009). Assim, equipados com dispositivos de computação, usamos a nossa inteligência para resolver problemas que não ousaríamos assumir antes da era da computação e construímos sistemas com funcionalidades limitada apenas pela nossa imaginação (Wing, 2006). A essência do pensamento computacional é pensar em dados e ideias, utilizando e combinando esses recursos para resolver problemas (Phillips, 2009). Os professores podem incentivar os alunos a "pensar computacionalmente" promovendo atividades onde os alunos para além de "utilizar" ferramentas de tecnologia e informações, passam a "criar" com ferramentas e informações (Phillips, 2009).

Tal como referido anteriormente na caracterização da disciplina de Aplicações Informáticas B, na perspetiva de Targetjobs (2007, citado por Carvalho et al., 2008), uma das competências mais valorizadas pelos empregadores é capacidade para resolver problemas, e de facto o pensamento computacional envolve a resolução de problemas, ajudando na transformação de um problema aparentemente difícil, para um que se sabe como resolver, através da redução, divisão, ou simulação (Wing, 2006). Segundo Wing (2006) no quotidiano este tipo de pensamento é usado várias vezes: quando um jovem decide o que colocar na sua mochila da escola, pode-se afirmar que utiliza “prefetching” para escolher o que colocar na mochila e faz “caching” ao armazená-las temporariamente na mochila. Quando alguém perde algo, realiza “backtracking” numa tentativa de encontrar o item, através da memória mental relembrando passo a passo o que fez ao longo do dia até ao ponto em que o perdeu (Wing, 2006). Escolher a fila de caixa de supermercado em que ficar, envolve pensar no modelo de performance de um sistema multi-servidor, ou seja comparando um multi-servidor ao conjunto de caixas e empregados existentes à saída do supermercado, a pessoa examina aquela que tem melhor desempenho por ser mais rápida (Wing, 2006). Assim, tal como a computação ubíqua era o sonho de ontem que se tornou realidade hoje, o pensamento computacional é a realidade de amanhã que será uma habilidade fundamental utilizado por todos no mundo a partir de meados do século XXI (Wing, 2006).

3.2. Definição de programação

Um programa pode ser definido como um conjunto de instruções, ou ordens dadas à máquina, com o fim de executar determinada tarefa (Aguilar, 2008). No entanto, para Gomes, Henriques e Mendes (2008) o ato de programar, vai muito além da escrita de um conjunto de linhas de código numa dada linguagem, consideram-no como uma arte e mesmo uma ciência. Acrescentando que a programação é uma arte, porque com criatividade existem várias formas diferentes de codificar instruções; e uma ciência, porque é constituída por um conjunto de regras orientadoras, onde é necessário usar de lógica e onde existem métodos rigorosos que asseguram a eficiência, economia e utilidade dos programas gerados (Gomes et al., 2008).

Além de transferência de competências, quando os alunos aprendem a programar adquirem uma sensação de domínio sobre um instrumento tecnológico ao mesmo tempo que estabelecem contato com conteúdos programáticos de diferentes disciplinas, tais como: a ciência, a matemática e a arte da construção de modelos intelectuais (Papert, 1980). Para Mulder (2002, citado por Saeli et al., 2011) a programação é um assunto de nova geração, que reúne peças de diferentes áreas, tais como: linguística, matemática e economia. Esta completude dá aos alunos a oportunidade de se deparar com um assunto multidisciplinar, que liga diferentes aspetos numa única disciplina (Mulder, 2002, citado por Saeli et al., 2011). Desta forma, os alunos podem aplicar conhecimentos adquiridos anteriormente, e em alguns casos aprofundá-los como fizeram Resnick e colaboradores (1990, citado por Saeli et al., 2011) que através da programação de robôs aprofundaram o conhecimento dos alunos sobre o conceito de física, atrito.

O processo de projetar programas passa por diversas fases (Aguilar, 2003): a análise onde o problema é analisado tendo em conta a especificação de requisitos; o projeto, uma vez analisado o problema, projeta-se uma solução algorítmica que resolva o problema; a implementação consiste na conversão da solução algorítmica numa linguagem de programação. A abordagem dividir-para-conquistar ajuda na fase de análise, através da divisão de um problema em vários subproblemas que são semelhantes ao problema original, mas menores em tamanho, onde a resolução dos subproblemas combinados criam uma solução para o problema original (Cormen, Leiserson, Rivest & Stein, 2009). Resumindo, esta abordagem envolve três passos: dividir o problema em vários subproblemas; conquistar os subproblemas resolvendo-os e reunir as soluções para os subproblemas na solução do problema original (Cormen, et al., 2009).

3.3. Conceitos de programação e suas vantagens

O domínio da programação transcende o conhecimento de técnicas de programação, englobando igualmente a compreensão profunda dos conceitos de computação (Martins & Cravo, 2011). Sebesta (2003) defende que o estudo de conceitos de programação é fundamental e beneficia os estudantes por aumentar a sua capacidade de expressar ideias. O autor (Sebesta, 2003) continua por declarar que a profundidade do nosso pensamento é influenciada pela expressividade da língua em que comunicamos em pensamento. Assim, o programador que apenas se foca na linguagem de programação em que está a desenvolver *software*, fica com um nível de compreensão limitado sobre a programação em si (Sebesta, 2003). Por outro lado, aqueles que apostam na compreensão de conceitos de programação melhoram a capacidade de escolha de linguagens apropriadas, porque ao ter noções de conceitos de programação e conhecer outros tipos de linguagens de programação, faz com que o programador não se feche apenas na linguagem que domina, estando assim em melhor posição para fazer escolhas apropriadas (Sebesta, 2003). Para além disso, o conhecimento de conceitos de programação aumenta a capacidade de aprender novas línguas, uma vez que como as linguagens de programação estão em constante evolução, é requerido ao programador uma atualização e aprendizagem constante e uma vez adquirida uma compreensão completa dos conceitos fundamentais da linguagem, torna-se muito mais fácil ver como esses conceitos são incorporados ao projeto da linguagem a ser aprendida (Sebesta, 2003).

3.4. Definição de algoritmo

A etimologia da palavra algoritmo provém de uma variação fonética da pronúncia do último nome do matemático árabe Abu Ja'far Mohammed ibu-Musa al-Khowarizmi, que viveu no século IX, e que desenvolveu um conjunto de regras para efetuar operações aritméticas com números decimais (Martins & Cravo, 2011). Existem várias definições para algoritmo, segundo Martins e Cravo (2011) informalmente, um algoritmo é uma sequência de instruções que podem ser executadas de modo a atingir um determinado objetivo, sendo que este objetivo, está sempre associado à solução de um problema. Os autores continuam, referindo que um algoritmo não representa, necessariamente, um programa de computador, mas sim os passos necessários para realizar uma tarefa a ser executada por um agente, o qual pode ser humano, mecânico, eletrónico, ou outro. Na

nossa vida cotidiana utilizamos algoritmos com muita frequência, por exemplo, as instruções de uma receita de culinária, as instruções para fazer um pagamento via multibanco, as instruções para montar equipamentos, ou as instruções para nos deslocarmos a determinado local (Martins & Cravo, 2011). O conceito de algoritmo é fundamental em Informática, sendo que os computadores só fazem aquilo que mandamos, e não necessariamente o que desejamos que eles façam (Martins & Cravo, 2011). Nesta perspectiva, alguns encaram a Informática como o ramo da ciência que estuda o modo de exprimir “como”, ou seja, em Informática e em particular em programação estamos interessados em desenvolver sequências de instruções que dizem ao computador quais os passos a seguir para resolver problemas, especificando concretamente como resolvê-lo (Martins & Cravo, 2011).

Embora um algoritmo não seja mais do que uma sequência de passos a seguir para atingir um determinado objetivo, nem todas as sequências de passos para atingir um dado objetivo podem ser consideradas algoritmos (Martins & Cravo, 2011). Para isso é necessário que sejam reunidas três características a nível de rigor, eficácia e finitude. Segundo Martins e Cravo (2011), um algoritmo tem que ser rigoroso, no sentido de cada instrução deve ser simples e sem ambiguidade, especificando exata e rigorosamente o que deve ser feito, sem margem para segundas interpretações. Tal como ser eficaz, ou seja, cada instrução do algoritmo deve ser suficientemente básica e bem compreendida, de modo a resolver um problema utilizando um número finito de passos (Martins & Cravo, 2011). Por último, um algoritmo deve ser finito, ou seja, o algoritmo deve levar a uma situação em que o objetivo tenha sido atingido e não existam mais instruções para ser executadas (Martins & Cravo, 2011). Existem três formas de descrever algoritmos, através da descrição narrativas, de pseudolinguagens e de linguagens gráficas (Martins & Cravo, 2011). A descrição narrativa consiste num método direto de expressar um algoritmo passo a passo verbalmente, enquanto a pseudolinguagem é uma linguagem de especificação de algoritmos, que consiste na escrita de instruções através de palavras semelhantes com o Inglês ou Português, que facilitam tanto a escrita como a leitura de programas (Martins & Cravo, 2011). Por último, as linguagens gráficas permitem uma representação gráfica de um algoritmo, sendo o mais conhecido o Diagrama de Fluxo. Os símbolos utilizados nesta representação foram padronizados pelo Instituto Norte Americano de Padronização [ANCI].

3.5 Problemática no Ensino-Aprendizagem de Programação

A programação é considerada difícil de aprender e é assombrada por elevadas taxas de insucesso escolar (Ala-Mutka, 2003; Gomes et al., 2008). O ato de ensinar programação por si só não é fácil, uma vez que devido a complexidades individuais, diferentes alunos revelam diferentes dificuldades a nível da programação (Saeli, et al., 2011). Na generalidade os alunos não têm motivação intrínseca para estudar programação, facto que é agravado quando não veem utilidade nos programas produzidos (Bereiter & Ng, 1991, citados de Gomes et al, 2008). Na perspetiva dos alunos programar é complexo, por vezes confuso e envolve muito código, o que resulta em dificuldades a nível da compreensão da sintaxe das linguagens de programação (Gomes, Martinho, Bernardo, Matos, & Abrantes, 2012). Para além disso, Ala-Mutka (2003) reforça que os alunos na programação tendem em falhar na correta aplicação de conceitos, compreendidos teoricamente. Estes factos vão de encontro a estudos como o de Kurland, Pea, Clement e Mawby (1989), onde constataram que mesmo depois de dois anos de instrução, o nível de compreensão de programação por parte dos alunos é baixo.

Apesar disso, autores como Papert (1980) defendem que o ensino de programação, quando suportado por ferramentas e estratégias de ensino adequadas, em certa medida pode fazer os alunos peritos. Aliando o fascínio que os alunos têm pelos robôs, com as suas preferências por atividades práticas ou *hands-on*, em detrimento de uma exposição de conteúdos excessiva, pensa-se que a robótica educativa consiga proporcionar contextos de aprendizagem interessantes em que os alunos percecionam a utilidade prática dos conceitos que estão a aprender (Gomes et al., 2012; Kumar & Meeden, 1998, citado de Koski, Kurhila & Pasanen, 2008). Com a utilização de linguagens de programação visual pretende-se minimizar as dificuldades dos alunos a nível de sintaxe. Por último, para ajudar os alunos a aprender tanto o conceito de programação como estratégias para a sua utilização, à semelhança do trabalho desenvolvido por Deek, Kimmel, e McHugh (1998, citado por Ala-Mutka, 2003) apresenta-se como solução uma metodologia de aprendizagem por problemas.

3.6 Robótica educativa

Os primeiros trabalhos desenvolvidos com robótica na educação foram iniciados por Seymour Papert (1980) no Laboratório de Inteligência Artificial do MIT, que durante as suas experiências percebeu que o computador atraía as crianças e isso poderia facilitar o processo de aprendizagem. Do ponto de vista industrial a robótica é o conjunto de conceitos básicos de mecânica, cinemática, automação, hidráulica, informática e inteligência artificial, envolvidos no funcionamento de um robô com a finalidade de produzir um trabalho de melhor qualidade, realizado num menor espaço de tempo e com menos gastos (Usatequi & Leon, 1986; Maliuk, 2009). Do ponto de vista educacional a robótica educativa pode ser definida como a utilização de conceitos da robótica industrial em um ambiente de aprendizagem, onde o aluno interage com o concreto e o abstrato para a resolução de problemas propostos (Usatequi & Leon, 1986; Maliuk, 2009).

Para Johnson (2003) os robôs tornam-se mais do que um simples brinquedo e mais do que uma simples moda, visto realizarem comportamentos mais interessantes do que apenas movimentos, podendo até mesmo ter algum nível de autonomia (Johnson, 2003). O mesmo autor (Johnson, 2003) considera o robô acima de educacional, é “*edutainment*” (p. 4), termo que sugere que as crianças podem ser educadas ao mesmo tempo em que se divertem, onde o ato de aprender torna-se uma atividade prazerosa. De facto, existe um grande potencial na robótica para motivar muitas partes do currículo, criando melhores currículos e currículos mais inspiradores (Lego, 2006; Johnson, 2003). Nesse sentido, as universidades têm maior liberdade para definir o seu próprio currículo de robótica, em contraste, a maioria das escolas não têm os recursos e a liberdade de o fazer, sendo que devem reger-se pelo currículo nacional (Johnson, 2003). Mas isto não quer dizer que se torna impossível introduzir a robótica educativa nos currículos escolares vigentes, isto porque uma das vantagens da robótica educativa é a sua multidisciplinariedade, uma vez que trabalha com conceitos de muitas disciplinas, incluindo as ciências de computação e mais especificamente a programação. Segundo Alimisis (2009) a robótica educativa pode ser classificada de duas formas, como ferramenta ou como objeto aprendizagem, de acordo com o papel que a robótica desempenhar no processo de aprendizagem. A robótica como ferramenta de aprendizagem, é quando o robô é proposto como uma ferramenta para o ensino e aprendizagem de outras disciplinas escolares, tais como matemática ou ciências (Alimisis, 2009). A robótica como objeto de aprendizagem, tem seu foco em

assuntos relacionados diretamente com a robótica em si, entre elas, a construção do robô, a programação do robô e a inteligência artificial (Alimisis, 2009).

A revisão de literatura sobre robótica educativa mostra que os robôs têm sido usados nos mais diversos níveis de ensino, desde o ensino básico ao superior. Em Portugal, no primeiro ciclo foram realizados estudos, tais como o projeto RobôCarochinha, (Ribeiro, 2006) onde através da programação e encenação da história da Carochinha os robôs são utilizados como ferramenta para fazer os alunos aprenderem matemática, estudo do meio, língua portuguesa e expressões, e educação tecnológica. No entanto, surgem projetos mesmo no primeiro ciclo onde o robô é usado como objeto de aprendizagem, com a finalidade de promover o pensamento computacional, acreditando que esta seja uma competência essencial e transversal a todas as áreas do saber (Serafini, 2011). Zilli (2004) no seu estudo sobre o uso da robótica educativa no ensino fundamental de 5ª a 8ª séries de Curitiba, afirma que a robótica educativa, além de proporcionar aos alunos o contacto com a tecnologia atual, sugere o desenvolvimento de competências, tais como: habilidades manuais e estéticas, raciocínio lógico, capacidade crítica e relações interpessoais e intrapessoais. Para além do que estimula a utilização de conceitos aprendidos em diversas áreas do conhecimento, o trabalho com pesquisa, a resolução de problemas, a aplicação de teorias em atividades concretas, e utilização da criatividade em diferentes situações (Zilli, 2004). Em Portugal tem sido feita investigação com resultados positivos referente às disciplinas de 10º ano do curso profissional de Gestão e Programação de Sistemas Informáticos Programação e Sistemas de Informação e à disciplina de 10.º ano do Curso Profissional de Informática de Gestão, Linguagens de Programação (Bernardo, 2012; Costa, 2012; Gomes, 2012, Varanda, 2012).

Referente ao ensino superior Kumar e Meeden (1998) têm usado os robôs em disciplinas de Inteligência Artificial, apontando como motivo o facto de os robôs fascinarem os alunos e esse interesse poder ser utilizado como um incentivo para a aprendizagem, visto que os alunos querem ver o sucesso da sua invenção. Imberman (2004) relata que depois de começar a usar robôs na sua disciplina de Inteligência Artificial a taxa de inscrição aumentou. Kumar (2001) usou robôs também numa disciplina de Inteligência Artificial e relata que quando os estudantes foram questionados se eles recomendariam a disciplina para os seus amigos, mais de 90% responderam que sim. Na Santa Clara University (SCU) a Robotic Systems Laboratory (RSL), promove um conjunto de projetos anuais em que equipas de alunos finalistas projetam e segundo os

resultados da avaliação final o desempenho dos estudantes nos projetos de robótica foi claramente superior aos dos participantes em projetos que não utilizam a Robótica. Klassner (2002) sobre a sua disciplina de ciências da computação, afirma que os alunos aprenderam conceitos fora do currículo, estavam mais confiantes. Além disso, os robôs trazem um fator de diversão para as aulas (Imberman 2004 e Kumar, 2001).

Benitti (2012) salienta que apesar de a robótica ter grande potencial para auxiliar o ensino, a aprendizagem dos alunos não é garantida apenas pela sua simples aplicação em contexto de sala de aula, existindo fatores que podem influenciar o seu sucesso, entre eles a escolha de um robô apropriado, o domínio do professor sobre a robótica, as salas de aula terem espaço suficiente para a experimentação e movimentação do robô, a aplicação de metodologias de aprendizagem adequadas e a boa organização das equipas de trabalho.

3.7 Linguagens de programação visual

As linguagens de programação podem ser divididas em quatro paradigmas: imperativo, lógica, funcional e orientada a objetos (Sebesta, 2003). A linguagem visual, constitui uma subcategoria das linguagens imperativas (Sebesta, 2003). As linguagens visuais são linguagens de programação de mais alto nível de abstração, que utilizam sistemas de representação de informação pictórica ou pictural, sendo de fácil utilização, devido a permitirem de uma forma simples através da interface gráfica gerar programas, através de um sistema de *drag-and-drop* (Sebesta, 2003).

Dentro destes paradigmas, a escolha da linguagem de programação a utilizar numa disciplina introdutória de programação é sempre um aspeto controverso. A sintaxe das linguagens imperativas, comumente ensinadas nos primeiros anos de programação, tais como Pascal e C, exigem um grande esforço na descrição da linguagem, o qual remete para segundo plano os conceitos essenciais subjacentes à programação (Boshernitsan & Downes, 2004). Os defensores das linguagens visuais referem que as linguagens de programação textuais têm provado ser bastante difíceis de aprender, porque na generalidade as pessoas pensam e lembram coisas em termos de imagens, relacionando-se com o mundo de uma forma inerentemente gráfica e fazendo uso das imagens como um componente fundamental do pensamento criativo, assim sendo, ao reduzir ou eliminar totalmente a necessidade de traduzir ideias visuais em representações textuais artificiais pode ajudar a atenuar este problema de aprendizagem (Boshernitsan & Downes, 2004).

3.8 Robô Lego® Mindstorms® NXT

A indústria LEGO® no seu vasto leque de produtos contém uma gama especial para as escolas e o ensino, denominada LEGO® MINDSTORMS® *Education*. Este *kit* possui um conjunto variado de peças denominadas *bricks* usados para a construção dos robôs (Lego, 2009) (Anexo D). Contem igualmente um *NXT Intelligent Brick* [i-Brick] com um microcontrolador de 32-bits e com módulo de comunicação por bluetooth, três servo-motores, dois sensores de toque, um sensor de som, um sensor ultrassónico e um sensor de luz e cor (Lego, 2009) (Figura 6). Para além disso, inclui sete cabos conetores RJ12 para ligar os motores e sensores ao i-Brick; um cabo USB para ligar o i-Brick ao computador; um CD com o *software* NXT-G; um guia de utilizador com instruções sobre *hardware*, *software* e construção de robôs e um Teste Pad para testar os robôs (Lego, 2009).



Figura 6. Motores, sensores e i-Brick do kit Lego Mindstorms NXT³

Considerando a realidade do sistema educativo e as suas dificuldades orçamentais, os robôs programáveis da Lego® em comparação com outros têm um preço razoável (Gaspar, 2007). Outra vantagem advém do facto deste *kit* ser compatível com outros materiais da LEGO®, como também ser possível adquirir conjuntos de peças suplementares em separado (Gaspar, 2007). Tal como, a facilidade de construção dos robôs, visto não exigir conhecimentos prévios ao nível da eletrónica, sendo a montagem muito simples através de encaixes. Gaspar (2007) afirma que o entusiasmo proporcionado pelo uso deste *kit* faz com que os alunos se tornem admiradores dos produtos Lego®.

³ Figura retirada de <http://mindstorms.lego.com>

3.8.1 Programação no i-Brick e NXT-G.

Oficialmente os robôs Lego® Mindstorms® NXT podem ser programados de duas formas: diretamente pelo i-Brick, ou através do *software* de programação NXT-G que vem juntamente com o *kit* (Lego, 2009). Com este robô não é preciso um computador para programá-lo, através do submenu do i-Brick NXT Program é possível criar milhares de programas (Lego, 2009) (Figura 7). Este tipo de programação está limitada a um total de máximo cinco passos e para cada um deles existe um leque determinado de comandos pré-definidos (Lego, 2009). A Figura 7 exemplifica alguém a programar diretamente no i-Brick, através do submenu NXT Program, e a selecionar para o primeiro passo, o símbolo de uma seta, correspondente ao comando Forward, que faz o robô andar indefinidamente para a frente.

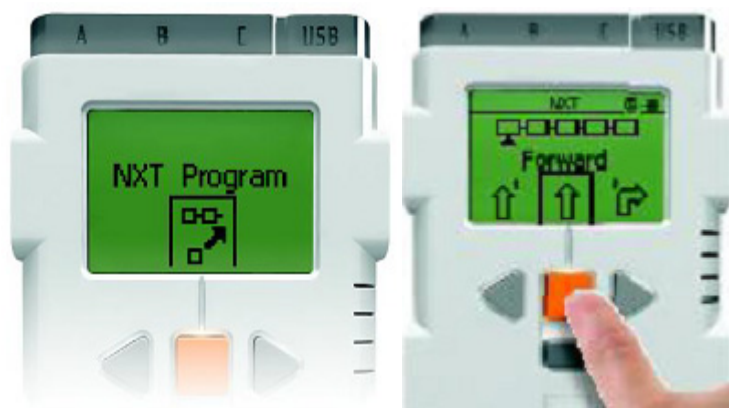


Figura 7. Programação direta no i-Brick

A programação através do NXT-G permite uma maior profundidade a nível de comandos e das suas propriedades. A Figura 8 mostra o ambiente do *software*, onde ao número um corresponde o separador ativo RoboCenter, que contém alguns manuais de construção de robôs (Lego, 2009). O número dois corresponde ao separador que neste caso não está ativo My portal, que estabelece ligação à internet possibilitando funcionalidades tais como *download* de *wallpapers* e som extras para o robô (Lego, 2009). O número três corresponde à Tool Bar, onde se podem encontrar comandos para abrir e guardar ficheiros, entre outros (Lego, 2009). O número quatro é a Work Area, onde se podem arrastar blocos formando programas, neste caso foi arrastado o bloco MOVE (Lego, 2009). O número cinco é a Little Help Area, que fornece dicas ao programador, por exemplo quando selecionado um bloco apresenta uma breve descrição

da utilidade do mesmo. O número sete corresponde à Programming Palette, existindo três e neste caso está selecionada a Common, nestas paletes pode encontrar-se os blocos (Lego, 2009). O número oito corresponde o Configuration Panel, que é ativado sempre que existe um bloco selecionado e permite alterar as suas propriedades (Lego, 2009). O número nove corresponde ao Controller que permite fazer o *download* dos programas para o robô (Lego, 2009). O número 10 corresponde ao botão superior esquerdo do Controller, chamado NXT Window que permite entre várias funcionalidades dar nome ao robô e visualizar se existem dispositivos Bluetooth conectados (Lego, 2009).

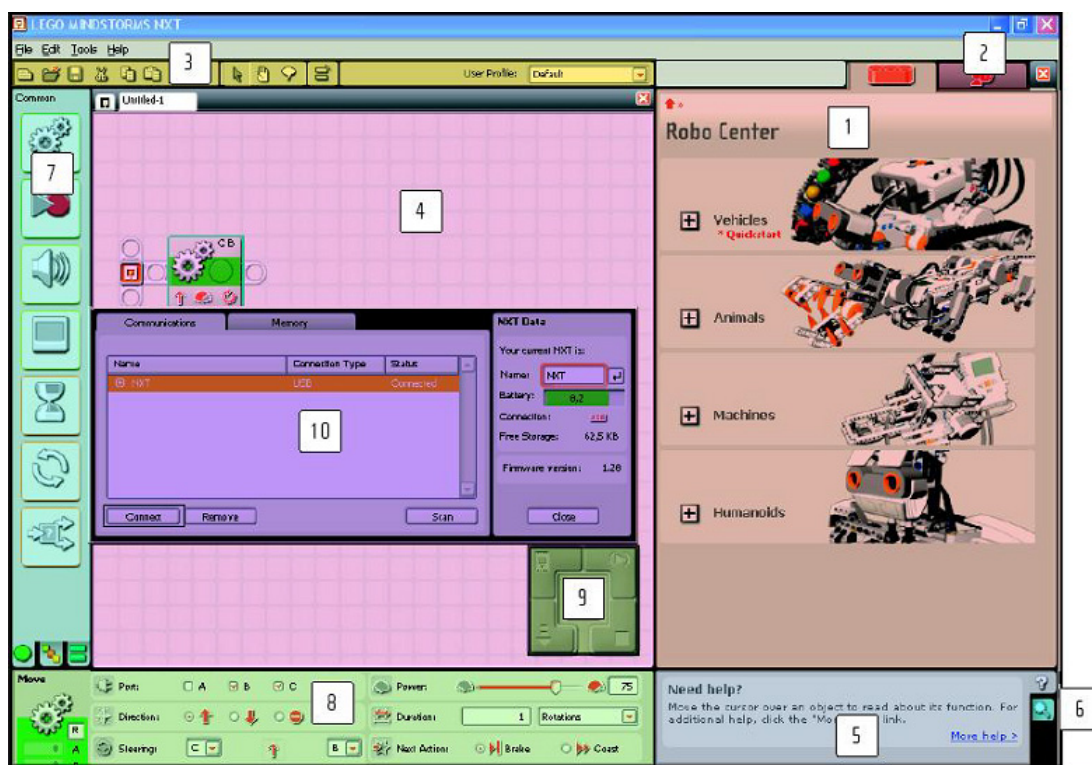


Figura 8. Interface gráfica do NXT-G

No entanto, é possível programar estes robôs em outras linguagens de programação não oficiais, como por exemplo Java através do *software* provido pela comunidade Lejos⁴, ou C# promovido pela comunidade NXT.NET⁵, entre outros. A adoção desta política de abertura, segundo Kim e Jeon (2007) tem como propósito dar a possibilidade de interessados e entusiastas por este robô aprofundem o seu conhecimento adequando-o às suas necessidades.

⁴ Acessível em <http://lejos.sourceforge.net/>

⁵ Acessível em <http://nxtnet.codeplex.com/>

3.8.2 Bloco MOVE.

O bloco MOVE (Figura 9) é o bloco mais usado para controlar o movimento do robô e permite controlar qualquer um dos três motores separadamente, dois dos motores conjuntamente, ou mesmo três motores ao mesmo tempo (Griffin, 2010). Estando cada motor associado a uma porta A, B ou C (Griffin, 2010). Através da propriedade “*Direction*” pode-se escolher se o robô anda para a frente, para trás, ou se deve parar o seu movimento e pela “*Steering*” pode-se fazer o robô andar para a direita ou esquerda (Griffin, 2010). O “*Power*” controla o quão rápido o motor roda e “*Next Action*”, permite controlar como termina o movimento do robô, se para imediatamente ou se acaba por si só (Griffin, 2010).



Figura 9. Propriedade duração do bloco MOVE

A configuração “*Duration*” controla por quanto tempo o movimento irá durar (Griffin, 2010). Existem quatro opções, como mostrado na Figura X: ilimitado, graus, rotações e segundos. A duração por graus e rotações permite definir quanto o motor vai rodar (Griffin, 2010). A única diferença entre as duas configurações é o valor a inserir (Griffin, 2010). Note-se que 360 graus correspondem exatamente a uma rotação do motor. Torna-se assim mais conveniente para longas movimentações usar rotações e para movimentações mais curtas graus, porque os números são mais fáceis de trabalhar (Griffin, 2010). Ao digitar números em graus, apenas se podem usar números inteiros, mas quando se utiliza rotações, pode usar números até três casas decimais (Griffin, 2010). No entanto, o bloco Move tem apenas precisão até um grau, portanto, usando rotações não resultará em maior precisão, mesmo que se insira números mais precisos (Griffin, 2010). A duração por segundos permite executar os motores por um período de tempo específico, podendo ser introduzido um valor até três casas decimais (Griffin, 2010). Regra geral, um bloco termina a sua função antes do próximo bloco na sequência feixe corridas, mas o bloco Move com a duração ilimitado é uma das exceções porque o movimento nunca vai terminar por si só, ou seja, os motores continuam em funcionamento até encontrar outro bloco MOVE, ou assim que uma condição de paragem

o exija (Griffin, 2010). Note-se que um bloco MOVE com duração ilimitada não anda ilimitadamente por si só (Griffin, 2010).

3.9 Contextos de aprendizagem

Ensinar envolve criar contextos onde se possa aprender e aprender consiste em explorar contextos que permitem construir saberes, práticas, culturas e relacionamentos; assim a função central da educação passa pelo *design* de contextos onde se possa aprender contextos de aprendizagem (Figueiredo & Afonso, 2006).

A framework filosófica de Figueiredo e Afonso (2006) sobre contextos de aprendizagem relaciona aprendente, conteúdo e contexto num evento de aprendizagem (Figura 10). O conteúdo traduz-se em informação estruturada e codificada, que pode ser transmitida oralmente pelo professor, através de materiais multimédia ou outros meios; e o contexto consiste no conjunto de circunstâncias relevantes para o aprendente construir o seu conhecimento (Figueiredo e Afonso, 2006). Por fim, um evento de aprendizagem pedagógica é uma situação em que o aluno aprende através de interações entre conteúdo e contexto, e onde as atividades são construídas como sequências de eventos de aprendizagem, e o esforço exigido por cada tarefa contextualizada desencadeia a reflexão do aluno sobre os conteúdos pedagógicos (Figueiredo & Afonso, 2006).

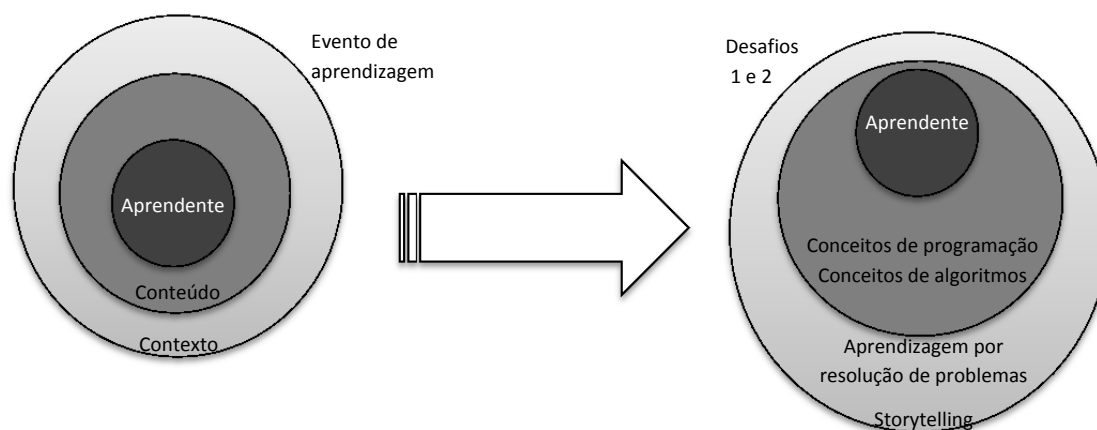


Figura 10. Modelo de Evento de Aprendizagem adaptado

Tendo como referência a framework filosófica de Figueiredo e Afonso (2006) para as cinco aulas de Prática de Ensino Supervisionada foram planificados dois eventos de aprendizagem, associados os conteúdos programáticos conceitos de programação e algoritmos. Quanto ao contexto de aprendizagem, entre as diversas práticas e técnicas pedagógicas contextuais optou-se pela aprendizagem por resolução de problemas e o *storytelling* (Figueiredo & Afonso, 2006).

3.10 Aprendizagem por resolução de problemas e storytelling

O problema é o ponto principal num processo de aprendizagem por resolução de problemas, servindo de estímulo para a aprendizagem, através da superação dos problemas e com vista a um único objetivo, o aluno deve identificar os conceitos implícitos, consciencializando-se sobre o que sabe e o que tem que saber, e depois apresenta soluções viáveis usando tanto a criatividade como o pensamento crítico, tornando-se ativamente envolvidos no processo de resolução do problema (Becker, 2009; Brownell & Jameson, 2004; Peterson, 2004; citados de Abreu & Loureiro, 2007). Esta é uma metodologia onde os alunos não são meramente recetores passivos, é uma pedagógica ativa, que visa os pressupostos de “learning by doing” ou “action learning” (Rodrigues, 2001 citado de Abreu & Loureiro, 2007), através do trabalho em equipa (Cégep, 2001 citado de Abreu & Loureiro, 2007). Este processo de questionamento e procura de soluções, conduz o aluno a tornar-se um agente ativo no seu processo de aprendizagem e indo de encontro à teoria construtivista, onde o conhecimento é entendido como uma ação do sujeito sobre realidade, conseguindo-se assim obter a aprendizagem (Papert, 1980).

Deste modo, o professor terá de formular problemas apropriados que levem os alunos a atingir os objetivos de aprendizagem esperados. Para isso, segundo Rusk e colaboradores (2008), em vez de se usar abordagens de *design* único, torna-se mais produtivo utilizar problemas com múltiplas perspetivas de desenvolvimento, associadas a um tema específico, onde cada participante pode trabalhar num projeto ao seu gosto, personalizando-o. Mais especificamente o *storytelling* permite envolver os alunos no processo de elaboração dos problemas, podendo mesmo cada um formular o seu próprio objeto de estudo.

Este tipo de metodologia vai de encontro com às finalidades e objetivos do programa curricular de Aplicações Informáticas B (Pinto, et al., 2009), que sugere que a disciplina deverá ter um caráter predominantemente prático e experimental, aliado a metodologias e atividades que invistam na aplicação prática e contextualizada dos conteúdos, a experimentação, a pesquisa e a resolução de problemas, desenvolvendo nos alunos o interesse pela pesquisa, descoberta e inovação e a capacidade de trabalhar em equipa.

4. Planeamento da Prática de Ensino Supervisionada

Com o ensino pretende-se alunos de “cabeça bem-feita”, não apenas de “cabeça bem-cheia”, ou seja, alunos que tenham a capacidade de transformar a informação em conhecimento pertinente que, segundo o autor, é o conhecimento que é capaz de situar informação no seu contexto (Morin, 2000, citado por Alarcão, 2000). Para isso, os professores têm que ser capazes de produzir currículos e correspondentes planificações, de modo a permitirem que os alunos sejam o centro do seu processo de aprendizagem e que consigam desenvolver sua capacidade reflexiva (Alarcão, 2000). A planificação de estratégias pedagógico-didáticas decorre da definição, prioridade e sequência de objetivos e conteúdos que complementam-se em processos e métodos de os ensinar ou aprender, de modo que o que se ensina ou aprende se vem a clarificar no como se ensina ou aprende, segundo um determinado tipo de avaliação (Ribeiro & Ribeiro, 1990).

Investigadores do Harvard’s Project Zero (Shotwell, Wolf, & Gardner, 1979, citado de Rusk et al., 2008) estudaram como as crianças interagem com os seus brinquedos, identificando dois estilos de crianças. As crianças “patterners” têm maior interesse no padrão e na estrutura do brinquedo, enquanto as “dramatists” preferem brincar fazendo dos brinquedos jogos de interação social. Segundo Rusk e colaboradores (2008) a robótica por si só em geral é estimulante para as crianças “patterners”, no entanto é possível entusiasmar igualmente as crianças “dramatists” contextualizando os desafios com temáticas, tais como histórias. Para Carvalho, Salles e Guimarães (2002) as histórias divertem, educam criando o desejo de continuar a aprender e a imaginação possibilita o desenvolvimento o pensamento criativo. Assim, o planeamento desta prática de ensino supervisionada foi projetado segundo uma temática de heróis robóticos e com um forte cariz de reflexão, onde os alunos foram incentivados a refletir sobre a sua experiência de aprendizagem. Tal como, um forte cariz prático de experimentação do robô, porque segundo Cook e Cook (1999) a capacidade de retenção de conhecimento dos alunos sobre aquilo que estão aprender experimentando ou *hands-on* é de 80%, e daquilo que explicam a outros é de 90%.

4.1. Temática geral e específica dos desafios

A temática geral da prática de ensino supervisionada assenta na proposta de planificação para a disciplina de Aplicações B de Ferreira, Oliveira, Celestino e Abrantes (2013, Maio), onde a robótica educativa foi incluída ao longo de todo o ano letivo. No primeiro período, às unidades de ensino-aprendizagem Conceitos básicos multimédia e Introdução à teoria da interatividade, os alunos criariam com base num guião histórias de heróis robóticos e a correspondente banda desenha, depois desenrolar-se ia um concurso na escola com auxílio de QR-codes para a escolha da melhor história e esta seria a história que os alunos passariam a experienciar nas fases seguintes. No segundo período, à unidade de ensino-aprendizagem Introdução à programação os alunos passam a viver a história, ultrapassando desafios por construir e programar robôs. No terceiro período, à unidade de ensino-aprendizagem Utilização dos sistemas multimédia, os alunos elaborariam vídeos da banda desenhada e experiência vivida que determina o desfecho da história, podendo estes vídeo serem posteriormente partilhados com a comunidade escolar.

Devido a uma questão de limitação de tempo, sendo que a prática de ensino supervisionada decorreu durante cinco aulas de noventa minutos, não foi possível testar a totalidade da proposta de planificação de Ferreira e colaboradores (2013, Maio), sendo posta em prática apenas a unidade de ensino-aprendizagem Introdução à Programação fazendo uso do robô como objeto de aprendizagem. Tendo em mente a importância que os contextos de aprendizagem têm em proporcionar significado aos programas produzidos pelos alunos permitindo-lhe assim perceber a utilidade da programação, foram planeados eventos de aprendizagem temáticos, baseados no trabalho anteriormente desenvolvido por Oliveira, Ferreira, Celestino, Ferreira e Abrantes (2012), que consiste num história de super-heróis “*NXT Heroes*”, onde os alunos tentam salvar a sua cidade que está em perigo, através da personificação e programação de heróis-robóticos. Com base num guião proporcionado aos alunos que participaram no estudo para o artigo (Oliveira et al., 2012), eles elaboraram uma história, descrevendo no primeiro capítulo (Figura 11) que tudo corria bem em Smallville até o dia em que Lektor se muda para a cidade. Este era um multimilionário que tinha a cidade no bolso, incluindo os super-heróis IronMan e Wolverine. Para obter total controlo da cidade idealizou um ataque! Roubar todos os doces e *fast-food* da cidade. A Mulher invisível que ia comprar doces com o seu sobrinho apercebe-se do ataque e chama o Hulk para a ajudar.



Figura 11. Capítulo 1 da história NXT-Heroes

No capítulo dois (Figura 12), o Hulk chega rapidamente ao local e revoltasse com os seus ex-colegas IronMan e Wolverin por se terem aliada a Lektor neste ataque. No entanto, ao tentar impedi-los é capturado e mantido cativo por IronMan, enquanto Wolverin continua o saque pela cidade.



Figura 12. Capítulo 2 da história NXT Heroes

No capítulo três (Figura 13), a Mulher Invisível face a este acontecimento esconde-se. Entretanto a comunicação social emite um apelo mundial para ajudarem a cidade. Apercebendo-se desta situação os donos da Lego® enviam kits da Mindstorms® NXT para as escolas. No seu espírito nobre e juvenil um conjunto de alunos, apesar de não terem treino decidem corajosamente colaborar, e constroem os novos heróis-robóticos. A Mulher Invisível assume o compromisso de ajudar estes alunos altruístas a treinar os seus robôs e prepara-los para a sua missão de salvar a cidade, através de dois desafios.



Figura 13. Capítulo 3 da história NXT Heroes

A partir daqui os alunos passaram a viver a história na forma de dois desafios e a criam o seu desfecho ao longo de cada aula, com base no seu desempenho. Cada um dos desafios é introduzido aos alunos igualmente em forma de banda desenhada. No Desafio 1 (Figura 14), a Mulher Invisível conduz os heróis robóticos para um lugar secreto para dar início à “Semana de Treino”, com a finalidade de ajudar os alunos a aprender como controlar os seus robôs através da programação no i-Brick, tal como a reconhecer as peças do robô com especial ênfase nos sensores.



A Mulher Invisível conduz os corajosos heróis para um lugar secreto, e dá início à semana de treino.

Heroes



Desafio nº 1: Aprender a controlar o Robô.

Cada Robô é controlado através do i-Brick, representado pela imagem no lado esquerdo. Para dares vida ao teu Robô basta precionar por alguns segundos o botão laranja do i-Brick.



Robots



Sensores e Servo-Motores

Ao brick podem ser ligados 4 sensores diferentes e três servo-motores. Cada equipa deve descobrir quais os sensores que estão ligados ao i-Brick do seu Robô, e com eles cada elemento da equipa deve, criar o seu próprio programa, através do Menu: NXT PROGRAM.

No fim, a equipa em conjunto deverá escolher uma das soluções criadas e exemplificá-la com base na história elaborada interpretativa do comportamento do robô

Figura 14. Proposta Desafio 1

Existindo a possibilidade de os vilões poderem neutralizar os sensores dos robôs, no **Desafio 2** (Figura 15) a Mulher Invisível prepara os alunos para controlar os robôs em NXT-G sem usarem sensores, através do bloco MOVE. Assim, a Mulher Invisível informa os alunos que têm que conseguir entrar no castelo guardado pelo terrível Lama que se encontra na ilha secreta. Para completarem o desafio 2 os alunos terão de contornar três subdesafios distintos: i) percorrer o túnel para chegar à ilha, ii) avançar a muralha, e iii) imobilizar o Lama através de uma habilidade especial do robô para conseguirem entrar no castelo.



A Mulher Invisível quer preparar os heróis para todos os cenários possíveis. Imaginando que os vilões têm tecnologia que consiga neutralizar os sensores, os heróis têm que saber ultrapassar obstáculos sem eles.






As equipas têm de conseguir chegar ao castelo, que se encontra numa ilha secreta.





À entrada do castelo encontra-se o terrível Lama



Desafio nº 2:
Programar o Robô sem usar sensores em NXT-G.

Cada equipa terá como objetivo chegar à torre, desviando-se dos projeteis do Lama.





As equipas deverão:

1. Participar na elaboração da história;
2. Definir uma estratégia;
3. Elaborar pseudocódigo;
4. Criar programas em NXT-G;
5. Efetuar registo de vídeo

Figura 15. Proposta Desafio 2

4.2. Plano geral da prática de ensino supervisionada

Os professores planificam para diferentes períodos de tempo, Arends (2008) propõe cinco fases de planificação: anual, por período, por unidade de ensino-aprendizagem, semanal e diária. Para além disso, o mesmo autor (Arends, 2008) afirma que as planificações são um processo contínuo e cíclico, uma vez que os planos realizados para um dia específico são influenciados pelo que aconteceu na aula anterior e irão influenciar os planos para as aulas seguintes.

Em relação à prática de ensino supervisionada a professora cooperante mostrou disponibilidade para a realização da prática de ensino supervisionada no início do segundo período à unidade de ensino-aprendizagem Introdução à Programação. Havendo conhecimento por parte do professor orientador e cooperante da intenção de aplicar um método de ensino contextualizado e sob a temática de uma história de heróis robóticos, achou-se benéfico introduzir a temática logo no início da unidade de ensino-aprendizagem existindo a possibilidade de continuá-la ao longo de todo período letivo. Nesta fase a planificação anual, por período e por unidade de ensino-aprendizagem já haviam sido elaboradas pela professora cooperante, ficando a responsabilidade realizar as planificações semanais e diárias correspondentes às cinco aulas de duração da prática de ensino supervisionada. O plano a curto prazo da prática de ensino supervisionada (Anexo E) foi elaborado tendo em consideração a planificação da professora cooperante onde os conteúdos abordados nas primeiras aulas seriam conceitos de programação e algoritmos. Para além disso, no plano de cada aula foi tida em consideração as preferências dos alunos por atividades práticas (Gomes et al., 2012), assim em cada uma pelo menos 50% de tempo foi reservado para exploração dos desafios com os robôs. À semelhança do defendido por Cook e Cook (1999) que os alunos retêm 90% daquilo que explicam a outros, os momentos mais teóricos da intervenção letiva foram transformados em reflexões, onde se prende que haja o máximo interação com os alunos através de métodos de questionamento, envolvendo assim os alunos no assunto abordado.

O Quadro 4 sintetiza o planeamento das aulas realizado para a prática de ensino supervisionada com a duração de cinco aulas de 90 minutos, o correspondente a três semanas. A turma de Aplicações Informáticas B tinha aulas às segundas e quintas-feiras e a intervenção letiva iniciou-se a uma segunda-feira. Assim, a primeira semana teve como finalidade introduzir os alunos ao conceito de programação, a segunda semana teve como finalidade continuar o ensino e aprendizagem de conceitos de programação,

acrescentando o conteúdo de algoritmos e a terceira e última semana, correspondente apenas uma aula foi deixada em aberto, esta serviria para consolidação de conhecimentos.

Quadro 4. Síntese do plano a curto prazo da prática de ensino supervisionada

Semana 1		Semana 2		Semana 3
Conceitos de programação		Conceitos de programação e algoritmos		-
Aula 1	Aula 2	Aula 3	Aula 4	Aula 5
Reflexão Inicial sobre conceitos de programação	DESAFIO 1 – parte 2	Reflexão Inicial sobre conceitos de algoritmos	DESAFIO 2	Reflexão sobre a intervenção letiva
DESAFIO 1 – parte 1	Reflexão Final sobre conceitos de programação	DESAFIO 2	Reflexão Final sobre conceitos de programação e algoritmos	-

A primeira e segunda aulas foram reservadas para o Desafio 1. A parte inicial da primeira aula foi reservada para uma reflexão conjunta sobre conceitos de programação e o restante da aula foi reserva para experimentação do robô através do Desafio 1. Na segunda aula os alunos continuam a vivência da história através dos robôs durante metade do tempo de aula, enquanto o restante tempo final foi reservado para uma reflexão final conjunta sobre que conceitos de programação os alunos haviam aprendido com o Desafio 1. A terceira e quarta foram reservadas para o Desafio 2. A terceira aula inicia-se como uma reflexão sobre conceitos de algoritmos tendo por base o trabalho desenvolvido no Desafio 1, a metade da restante aula foi reservada para experimentação do Desafio 2, onde os alunos teriam que aplicar os novos conceitos. Na quarta aula os alunos inicialmente continuam a exploração do Desafio 2, e a metade final da aula fica reservada para um momento de reflexão final sobre os conceitos de programação e algoritmos que os alunos aplicaram na vivência do Desafio 2. A quinta aula, dependerá do decorrer das aulas anteriores. Esta foi pensada para servir de momento de reflexão sobre o decorrer na intervenção letiva, podendo existir momentos de esclarecimento de dúvidas, tanto do Desafio 1 como do Desafio 2.

A planificação apresentada não é fixa ou inalterável, a cada aula tal como sugere Arends (2008) o professor deve refletir sobre quatro questões e ajustar a sua planificação sempre que achar necessário: Até que ponto o plano contém conteúdos que valem a pena reter para além da aula? Até que ponto, a planificação vai de encontro com a disciplina? Até que ponto os alunos têm conceções erradas sobre os conteúdos, ou acham difícil aprende-los? Até que ponto a planificação tem potencial para atrair os alunos?

4.3. Objetivos de aprendizagem

Na perspectiva de Ribeiro e Ribeiro (1990) “os objetivos são resultados de aprendizagem visados” (p.87), no sentido que todo o processo de ensino e aprendizagem move-se em direção do cumprimento de objetivos de aprendizagem que se pretende que resultem em competências ou habilidades adquiridas pelos alunos (Ribeiro & Ribeiro, 1990). A formulação de objetivos de aprendizagem é importante porque situa o aluno em relação às expectativas que existem sobre a aprendizagem e por outro lado torna a avaliação da aprendizagem pretendida pelo professor mais clara e evidente (Ribeiro & Ribeiro, 1990). Para Ribeiro e Ribeiro (1990) não se devem categorizar os objetivos como gerais ou específicos, salientando antes que os objetivos podem ser mais gerais ou mais específicos, podendo um objetivo geral ser dividido em vários objetivos mais específicos e assim consecutivamente.

Os objetivos de aprendizagem da prática de ensino supervisionada foram elaborados tendo com conta os eventos de aprendizagem Desafio 1, Atividade Extra e Desafio 2. Seguindo a sugestão de Rusk e colaboradores (2008) de que é vantajoso serem os próprios alunos a desenvolverem os seus contextos de aprendizagem, os desafios 1 e 2 estão associados a uma temática de heróis robóticos, no entanto o contexto de aprendizagem em si foi deixado em aberto, para que os alunos elaborassem o seu próprio problema base. Mas uma vez que segundo a caracterização da turma estes alunos não tinham experiência de programação e não conheciam ainda as funcionalidades do robô da Lego® Mindstorms® NXT, seria difícil para eles inicialmente pensar num problema base para posterior experimentação. Por isso, optou-se no Desafio 1 os alunos criarem o seu próprio contexto de aprendizagem, correspondente ao problema base, depois de experimentarem a programação e interpretação do comportamento do robô. Depois desta experiência os alunos para o Desafio 2 já estariam mais preparados para pensar e elaborar os seus contextos de aprendizagem logo de início.

O Desafio 1 foi dividido com duas partes. A parte 1 de Desafio 1 contém uma forte componente prática e tinha como finalidade familiarizar os alunos ao robô, fazendo-os compreender como o robô se movimenta, como funcionam os sensores e conhecer os comandos que os acionam. De modo a obter uma rápida familiarização com o robô optou-se pela programação direta no i-Brick, uma vez que dispõem de um conjunto de comandos pré definidos que resumem bem as potencialidades do robô. Conforme os alunos iam programando, deveriam interpretar do comportamento do robô o que os

ajudaria a compreender os programas produzidos e serviria como uma introdução aos algoritmos. Assim, o Desafio 1 parte 1 correspondente à primeira aula de intervenção letiva teve como objetivos de aprendizagem (Anexo F):

- Desenvolver programas no robô através do i-Brick;
- Interpretar o comportamento de robô.

A parte 2 do Desafio 1 contém uma forte componente de caráter reflexivo, tendo como finalidade fazer os alunos refletirem sobre os conceitos de programação envolvidos no trabalho realizado na parte 1 do desafio e introduzidos pela reflexão inicial sobre conceitos de programação ocorrida na primeira aula. Os alunos teriam de evidenciar que compreenderam os conceitos de programação envolvidos no seu trabalho, definindo o conceito de programa e identificando como aplicaram concretamente o conceito enunciado. Visto a programação no i-Brick apenas permitir a elaboração de programas com cinco instruções, para que os alunos não ficassem limitados a nível do conhecimento das funcionalidades gerais do robô, conhecendo apenas àquelas que usaram, foi acrescentado o objetivo de aprendizagem Listar os comandos disponíveis no i-Brick. Para além disso, os alunos teriam que identificar os sensores que utilizaram e explicar como interagiram com eles. O objetivo de aprendizagem Elaborar uma história, foi de encontro à sugestão de Rusk e colaboradores (2008), onde os alunos fazem um evento de aprendizagem criam o seu próprio contexto de aprendizagem, dando sentido e significado à sua experiência de aprendizagem através de uma pequena história. Desta forma, os programas passam a ser personalizados pelos alunos e mais originais, sendo este objetivo de aprendizagem uma introdução à capacidade dos alunos de pensar, identificar e elaborar problemas, capacidade esta imprescindível tanto na programação como em outras áreas do saber. Ou seja, esta história poderia ter sido o problema inicial dado aos alunos para concretizar a nível de programação, mas neste caso os alunos construíram o seu problema base depois da experimentação. Em suma, o Desafio 1 parte 2 correspondente à segunda aula de intervenção letiva tem como objetivos de aprendizagem (Anexo F):

- Compreender conceitos de programação:
 - Definir o conceito de programação;
 - Identificar o conceito de programação;
 - Listar os comandos disponíveis no i-Brick;

- Identificar sensores do robô e suas funcionalidades;
- Elaborar uma história.

Durante a reflexão inicial de conceitos de algoritmos foi promovida uma atividade extra, que embora estivesse relacionada com o Desafio 1, não fez parte integrante deste, com o objetivo pedagógico: Compreender conceitos de algoritmos. Esta atividade pretendeu, com base no programa elaborado em equipa fazer os alunos refletir sobre as características dos algoritmos e avaliar se a interpretação do comportamento do robô que fizeram poderia ser considerada um algoritmo ou não. Para a interpretação do comportamento do robô ser considerada um algoritmo, teria que ser rigorosa, eficaz e finita.

No Desafio 2 o contexto de aprendizagem foi desenvolvido pelos alunos no início do desafio. Visto neste caso a elaboração ser feita em conjunto com toda a turma, este não foi considerado a nível de objetivos de aprendizagem. O Desafio 2 teve como finalidade proporcionar aos alunos um contato mais abrangente com a multidisciplinaridade da robótica e da programação. Para além da disciplina de Português através da elaboração de histórias, foram incluídos conceitos das disciplinas preferidas dos alunos Matemática e Física, concretizados nos recursos: fita métrica para realizarem conversões e saco de pedras para equilibrar o peso do robô. No entanto, o foco do desafio continua a ser a aprendizagem de programação, por isso objetivos de aprendizagem são:

- Desenvolver programas em NXT-G utilizando o bloco MOVE;
- Elaborar pseudocódigo.

Depois de ultrapassado o Desafio 1 e estando agora os alunos familiarizados com os componentes e funcionalidades gerais do robô, seguindo uma metodologia por resolução de problemas tornou-se necessário aumentar o grau de complexidade do Desafio 2, passando-se assim à programação em NXT-G que permite um maior controlo do movimento do robô, através do bloco MOVE. Ao ser introduzido o conceito de algoritmo, através da atividade extra, neste desafio pretende-se que os alunos deixem de interpretar o comportamento de robô e passem a elaborar pseudocódigo, respeitando as suas características de rigor, eficácia e finitude.

4.4. Plano de Avaliação

O termo avaliação provoca regra geral nos alunos sentimentos de dificuldade, *stress* e ansiedade, estes associam-no a um momento decisivo, que se traduz numa nota quantificadora que define o seu sucesso ou insucesso escolar (Santos & Pinto, 2003). Ao abrigo do n.º 6 do artigo 12.º do Decreto-Lei n.º 6/2001, de 18 de Janeiro consta que “a avaliação é um elemento integrante e regulador da prática educativa”. Segundo o National Council of Teachers of Mathematics (citado por Santos, 2003) “a avaliação não deve apenas ser feita *sobre* os alunos, mas deve também ser feita *para* os alunos, para os guiar e potenciar a sua aprendizagem”. A avaliação passa assim a ser um dispositivo regulador da aprendizagem que tem como objetivo produzir melhorias e não punir (Santos, 2003). Ou seja, a avaliação deixa de prefigurar um momento isolado, e passa a ser o reflexo de um trabalho contínuo, um conjunto de momentos de avaliação que não servirão para quantificar o aluno, pelo contrário, serão momentos onde lhe é dada oportunidade de melhorar e aprimorar o seu trabalho (Santos, 2003). O aluno tem oportunidade assim de se ir definindo e desenvolvendo, ao ritmo da sua própria necessidade educativa (Santos, 2003). A avaliação passa a ser quantificada não em função do que o aluno acerta ou erra, mas na capacidade que evidenciou em corrigir-se e em reagir aos estímulos promovidos pelo feedback do professor (Santos, 2003). Passaremos assim para um método de avaliação, onde a essência reside na aprendizagem, onde os alunos não têm medo de errar, não têm medo de aprender e não tem medo de ser avaliados (Santos, 2003).

Nesse seguimento e em conformidade com critérios de avaliação da disciplina de Aplicações Informáticas B da Escola Secundária de Camões (Anexo G), a avaliação final da prática de ensino supervisionada é composta 90% pelos conhecimentos, competências e capacidades e 10% pelas atitudes e comportamentos (Quadro 5). As atitudes e Comportamentos são compostas por doze parâmetros. Para facilitar a avaliação dos mesmos, estes foram agrupados em três momentos específicos, reflexões sejam elas iniciais o finais, desafios tanto 1 como 2 e comportamentos durante o decorrer da aula (Anexo H). Durante as reflexões os alunos seriam avaliados aos parâmetros: atenção, sentido de oportunidade e espírito crítico. Enquanto nos desafios os alunos seriam avaliados aos parâmetros: atenção, interação, espírito iniciativa e cumprimento de tarefas. A nível de comportamento durante o decorrer na aula os alunos seriam avaliados aos parâmetros: respeito, responsabilidade, utilização correta dos equipamentos e respeito pelas normas da sala, pontualidade. Em relação ao critério material necessário este foi

desconsiderado em termos de avaliação, uma vez que nesta intervenção específica todos os materiais necessários seriam fornecidos aos alunos.

Quadro 5. Plano de avaliação final da prática de ensino supervisionada

Avaliação Final (100%) = Conhecimentos Competências e Capacidades (90%) + Atitudes e Comportamentos (10%)

Conhecimento Competências e Capacidades (90%) = Desafio 1 (45%)
+ Atividade Extra (10%) + Desafio 2 (45%)

Atitudes e Comportamentos (10%) = Pontualidade, apresentação do material necessário, atenção nas aulas, sentido de oportunidade, espírito crítico, interação, espírito de iniciativa, cumprimento das tarefas, respeito, responsabilidade, utilização correta dos equipamentos e respeito pelas normas de funcionamento das salas.

Em relação aos conhecimentos competências e capacidades dos alunos, estes ao longo da intervenção letiva foram avaliados através dos Desafios 1, 2 e atividade extra. Mais especificamente, durante a aula 1 e 2 os alunos teriam oportunidade de melhorar o Desafio 1 quer à componente individual, quer à de equipa. Os períodos de tempo em que foi possibilitado a melhoria dos desafio com auxilio de feedback do professor foram considerados como avaliação formativa segundo o ponto de visto de Santos (2003) enunciado anteriormente, e tal como especificado serviu para avaliar os alunos em termos de atitudes e comportamentos. O resultado final entregue na segunda aula foi avaliado em termos de conhecimentos, competências e capacidades através dos objetivos pedagógicos definidos. O mesmo ocorreu com o Desafio 2, durante as aulas 3 e 4 os alunos teriam a oportunidade de melhorar o desafio e na aula 4 os alunos teriam de entregar a versão final do desafio. A partir do momento em que os alunos entregam o resultado final do seu trabalho quer do Desafio 1 ou 2, a avaliação resultante será quantificada numa classificação, traduzindo-se num momento de avaliação sumativo, contando para a componente de avaliação conhecimentos. A classificação desses momentos de avaliação sumativos obedeceu a um conjunto de critérios.

Em relação ao Desafio 1 os alunos seriam avaliados 50% pelo trabalho desenvolvido individualmente e 50% pelo trabalho desenvolvido em equipa (Quadro 6). Os alunos seriam avaliados pelo seu trabalho desenvolvido individualmente como programadores, segundo os parâmetros: programação e interpretação do comportamento

do robô. O parâmetro programação correspondente ao objetivo de aprendizagem Desenvolver programas no robô através do i-Brick, incluiu facto de os alunos terem registado em vídeo o comportamento do Robô (25%), por terem usado sensores (25%), por não utilizarem empty's (25%) e pela originalidade dos seus programas, considerando original um programa que não seja exatamente igual ao de um elemento da mesma equipa (25%). Referente ao parâmetro interpretação, correspondente ao objetivo de aprendizagem Interpretar o comportamento de robô e inclui a correta interpretação do comportamento do robô registado anteriormente em vídeo. Os alunos serão avaliados pelo seu trabalho desenvolvido em equipa, segundo o parâmetro reflexão, que inclui as equipas identificarem o conceito de programação (20%), conseguirem de forma clara evidenciar que aplicaram conceitos de programação no seu trabalho (20%), identificar das funcionalidades do i-Brick (20%), identificar dos sensores que utilizaram e suas funcionalidades (20%) e criar uma história interpretativa do comportamento do robô (20%). De modo a reduzir a ambiguidade na avaliação foram elaboradas algumas regras para a avaliação dos parâmetros (Anexo I e J).

Quadro 6. Plano de avaliação do Desafio 1

Desafio 1 = Trabalho Individual (50%) + Trabalho de Equipa (50%)

Trabalho Individual = Programa (50%) + Interpretação (50%)

Programação = Registo de vídeo (25%) + Uso de sensores (25%) + Não uso de empty's (25%) + Originalidade (25%)

Interpretação = Interpretação passo 1 (20%) + Interpretação passo 2 (20%) + Interpretação passo 3 (20%) + Interpretação passo 4 (20%) + Interpretação passo 5 (20%)

Trabalho em Equipa = Reflexão (100%)

Reflexão = Conceitos de programação (20%) + Aplicações de Conceitos de programação (20%) + Identificação das funcionalidades do i-Brick (20%) + Identificação dos sensores (20%) + História (20%).

Para a avaliação do objetivo de aprendizagem Compreender conceitos de algoritmos, promovido pela atividade extra integrada na reflexão inicial sobre conceitos de algoritmos, os alunos serão avaliados pelo correto julgamento sobre o seu trabalho

desenvolvido em equipa no desafio 1, segundo os parâmetros: rigor, eficácia e finitude. Assim, para os três parâmetros a cotação foi repartida em 5% por afirmarem se a sua interpretação do comportamento do robô é rigorosa, eficaz e finita e 35% para justificarem a sua opção à exceção da justificação de finitude que vale 15% (Quadro 7).

Quadro 7. Plano de avaliação da atividade extra

Atividade Extra = Rigor (5%+ 35%) + Eficácia (5%+35%) + Finitude (5% + 15%)
--

Por último, no Desafio 2 os alunos seriam avaliados pelo trabalho desenvolvido em equipa. Este desafio foi idealizado para conter três fases: i) subir a rampa, ii) avançar a muralha e iii) habilidade especial. Em cada uma destas fases os alunos foram avaliados segundo os objetivos de aprendizagem Desenvolver programas em NXT-G utilizando o bloco MOVE e Elaborar pseudocódigo, segundo os parâmetros programa (40%), registo de vídeo (20%) que servira de comprovativo da concretização correta do programa e o pseudocódigo (40%) (Quadro 8).

Quadro 8. Plano de avaliação do Desafio 2

Desafio 2 = Fase 1 (40%) + Fase 2 (40%) + Fase 3 (20%)

Fase N = Registo Vídeo (20%) + Programa (40%) + Pseudocódigo (40%)

4.5 Recursos e materiais

Os recursos e materiais mobilizados para esta intervenção letiva tiveram como finalidade ajudar os alunos no cumprimento dos objetivos de aprendizagem segundo os parâmetros de avaliação definidos, tendo em vista auxiliar uma atividade específica.

Quadro 9. Listagem de recursos e materiais para a prática de ensino supervisionada

Recurso/Material	Atividade	Descrição/Função	Anexo
<i>Moodle</i>	Geral	Usados para disponibilização de recursos	K
Computadores com ligação à Internet	Geral	Usados tanto pelo professor como pelos alunos para auxílio das atividades	-

Vídeo projetor	Reflexões	Para auxiliar a comunicação entre professor aluno	-
Apresentação sobre Conceitos de programação	Reflexão inicial sobre conceitos de programação	Para promover a comunicação entre professor aluno e estimular a aprendizagem	M
Apresentação sobre Conceitos de algoritmos	Reflexão inicial sobre conceitos de algoritmos	Para promover a comunicação entre professor aluno e estimular a aprendizagem	N
Folha de reflexão sobre algoritmos	Atividade Extra	Para auxiliar os alunos na reflexão das características dos algoritmos	O
6 Robôs	Desafio 1 e 2	Robôs da Lego Mindtorms	P
4 Camaras de vídeo	Desafio 1 e 2	Registo do comportamento do robô	-
NXT Heroes	Desafio 1 e 2	Introdução aos desafios e registo de trabalhos	K
Envelopes	Desafio 1 e 2	Usados para distribuição dos robôs e atribuição de características	R
Pen Drive	Desafio 1 e 2	Registo dos trabalhos dos alunos	-
Vídeo NXT Heores	Desafio 1 e 2	Introdução á temática dos desafios	-
Folha de testes no i-Brick	Desafio 1	Para auxiliar os alunos nos testes do robô, registrando os comandos e comportamento do robô	S
Caixas de cartão	Desafio 2	Para criação do cenário	-
3 Fitas métricas	Desafio 2	Para medir a distância percorrida numa rotação e medir a distância que se pretendia que o robô percorre-se	-
Sacos de pedras	Desafio 2	Usados para equilibrar o peso do robô ao subir a rampa	-

4.6. Estratégias de Ensino

O termo estratégia tem a sua origem na antiga Grécia, mais propriamente na linguagem militar, no sentido da ação militar, evidenciando a capacidade de conceber a ação na sua globalidade, nas articulações e sequências dos diversos passos, momentos e ações (Roldão, 2009). Uma estratégia em contexto educacional não é apenas uma atividade ou tarefa mas sim um “conjunto de ações do professor ou do aluno orientadas para favorecer o desenvolvimento de determinadas competências de aprendizagem que se têm em vista” (Vieira e Vieira, 2005, citados por Roldão, 2009, p.71). Uma estratégia

pode ser dividida em sub-estratégias que envolvem o recurso a diversas técnicas de ensino, concretizadas em atividades organizadas. Para elaborar uma estratégia torna-se importante explicitar com clareza os resultados da aprendizagem esperados, traçar percursos, monitorizá-los e verificar se os resultados estão a ser alcançados, e sempre que necessário rever as estratégias de ensino programadas (Ministério da Educação, 2009; Roldão, 2009).

Na reflexão inicial sobre Conceitos de programação, o professor através de questões realizará uma avaliação diagnóstica com o intuito de perceber o que os alunos sabem sobre programação. O professor incentivará a participação dos alunos e partindo das respostas dadas, mesmo as mais simples, completará o seu conhecimento através de dicas ou perguntas reflexivas. Para ajudar nesta tarefa o professor terá ao seu dispor o recurso Apresentação de conceitos de programação (Anexo L). Seguidamente o professor introduzirá o Desafio 1 através do visionamento do recurso Vídeo da história NXT Heroes, fará a distribuição dos robôs através do sorteio de envelopes e disponibilizará o recurso NXT Heroes (Anexo P). No Desafio 1 parte 1, os alunos serão organizados em equipas apesar do trabalho proposto ser de concretização individual. Os alunos, enquanto um dos elementos das equipas assumir o papel de programador e estiver em fase de experimentação do i-Brick, devem designar outro elemento para ser o seu ajudante, que deverá preencher a Folha de Teste no i-Brick (Anexo R), segundo as indicações do aluno programador. Perto do fim do tempo de experimentação, o ajudante deverá registar em vídeo o comportamento do robô e entregar o Folha de Teste no i-Brick ao aluno programador para que este proceda ao registo da sua experiência no recurso NXT Heroes slide 3, anexando um *hiperlink* para o vídeo do robô e interpretando o comportamento do robô nesse vídeo. Os restantes elementos da equipa deverão assumir o papel de secretários preenchendo os requisitos do slide 2. Durante esta fase de experimentação, quando questionado com dúvidas dos alunos o professor antes de fornecer qualquer sugestão ou orientação, tentará o máximo possível promover a capacidade de reflexão dos alunos através de questões, tais como: O que esperavas que o robô fizesse? Foi mesmo isso que programas-te? E o que aconteceu quando testaste o programa? Vamos rever os passos do programa um a um e testar o robô para detetar o problema? O que achas que é o problema? É importante que nesta fase inicial o professor acompanhe este processo de reflexão promotor do pensamento computacional, de modo a os alunos a desenvolverem de forma sólida e com o tempo o evidenciem, assim quando questionarem novamente o

professor possam apresentar as suas dúvidas já respondendo às questões enunciadas, sem que seja necessário o professor abordá-las.

No Desafio 1 parte 2, os alunos para além de estarem organizados em equipas todas as atividades exigiam trabalho em equipa. Cada equipa deverá seleccionar um dos programados criados anteriormente por um dos elementos da sua equipa proceder a uma reflexão em equipa sobre conceitos de programação e evidenciar que usaram os usaram. Para além disso, evidenciar que aprofundaram o seu conhecimento em relação à programação direta no i-Brick, as equipas deveriam detalhar todos os comandos disponíveis no i-Brick para cada um dos cinco passos. As equipas deveriam ainda evidenciar que conheciam os sensores de utilizados no programa seleccionado e sabiam como interagir com eles. Por último, as equipas teriam de elaborar uma história baseada no programa seleccionado. Findo o Desafio 1 parte 2 o professor promoverá um momento de reflexão final conjunta com a turma, onde cada uma das equipas terá a oportunidade de mostrar o seu trabalho. Um dos elementos da equipa ficará encarregue de contar a história, enquanto outro elemento interagirá com o robô de modo a concretizar a história narrada. No final, os alunos poderão fazer perguntas ou comentários gerais sobre as histórias dos colegas.

Na reflexão inicial sobre conceitos de programação e algoritmos, o professor tendo como base essa experiência dos alunos completará o seu conhecimento com auxílio do recurso Apresentação de conceitos de algoritmos (Anexo M). Porque, apesar de aparentemente este parecer um conceito novo, os alunos para a Desafio 1 haviam interpretado o comportamento do robô, essa interpretação é muito semelhante à elaboração de algoritmos. A estratégia escolhida assenta numa reflexão em equipa sobre o programa seleccionado pela equipa no Desafio 1, de modo a avaliar se a interpretação que fizeram do comportamento do robô, pode ser considerada um algoritmo respeitando as características de rigor, eficácia e finitude (Anexo N). Esta capacidade reflexiva de avaliar o trabalho realizado pretende promover o pensamento computacional dos alunos.

O professor deve comunicar no início do Desafio 2 aos alunos, que neste desafio cada equipa irá deparar-se com obstáculos diferentes à concretização do desafio, visto que, para além de utilizarem robôs diferentes, iriam os controlar de forma diferente. Reforçando que o enriquecedor do desafio é cada um perceber as limitações e vantagens do seu robô. Depois o professor promoverá a criação da história base para o Desafio 2, onde os alunos com base na característica do robô contida no recurso envelope decide

como quer superar o desafio. A estratégia do desafio assenta na criação de programas em NXT-G sem fazer uso de sensores, visto o objetivo principal deste desafio ser os alunos aprenderem a controlar com precisão o movimento do robô através do bloco MOVE. Face ao enunciado do desafio 2 (Anexo J), cada equipa deverá completar os desafios propostos controlando a duração do movimento do robô de maneira diferente, seja por rotação, graus, tempo ou ilimitado. Será reservado tempo para que as equipas possam explorar o cenário e programar o robô, e interpretar o comportamento do robô. Na reflexão final sobre conceitos de programação e algoritmos o professor promoverá uma reflexão com a turma, onde cada equipa relatará a sua vivência do desafio. Aproveitando o contexto multidisciplinar do Desafio 2 será promovida uma reflexão entre os alunos sobre questões tais como: Tendo com conta os tipos de rodas usados em cada robô, qual seria o rápido? O mais estável? E dos três robô utilizados qual o mais indicado para superar o desafio? Entre outras. Para além disso, com a finalidade de reforçar o conhecimento adquirido pelos alunos em relação ao objetivo de aprendizagem Conhecer o bloco MOVE, o professor promoverá uma reflexão sobre o tipo de Duração utilizada por cada equipa, fazer os alunos perceber, neste tipo de cenários em que circunstâncias era mais vantajoso utilizar cada tipo de Duração. A conclusão resultará dos relatos dos alunos sobre as suas dificuldades e facilidades em cada uma da concretização das fases que compõem o desafio.

4.5. Método de trabalho

Uma equipa é um pequeno grupo de pessoas com habilidades complementares, assumindo papéis diferentes na realização de tarefas, mas comprometidos com um propósito comum (Foundation Coalition, 2001). Desenvolver habilidades de trabalho em equipa no ensino secundário aumenta o potencial dos alunos para a melhoria do desempenho académico e, simultaneamente, fornece habilidades importantes para prepará-los para o mercado de trabalho (Foundation Coalition, 2001). Tal como exemplificado na Figura 17, organizar os alunos em grupos não os conduz automaticamente a um desempenho superior (Foundation Coalition, 2001). Em vez disso, o desempenho das equipas está relacionado com eficácia na sua construção (Foundation Coalition, 2001). Por exemplo, nas pseudo-equipas, as pessoas designadas foram trabalhar na equipa, no entanto não têm qualquer interesse em trabalhar em conjunto,

levando-os a ser menos eficazes do que os indivíduos que trabalham separadamente (Foundation Coalition, 2001).

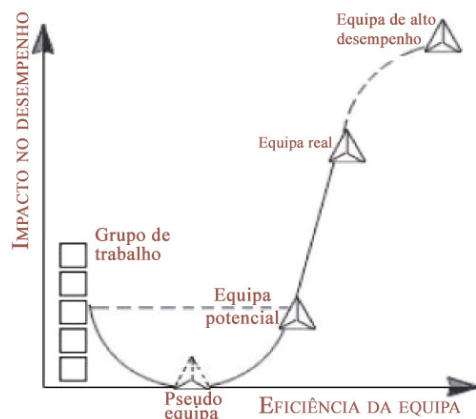


Figura 16. Curva de performance de equipas

Sendo que a escola tinha ao dispor 6 robôs para 22 alunos impunha-se a necessidade de agrupar os alunos de alguma forma. Assim, escolheu-se a organização por equipas assumindo os papéis rotativos de programadores, ajudantes ou secretários. O programador é o aluno que programa o robô, o ajudante auxilia o aluno programador e testa os seus programas, e o secretário empenha-se pelo registo das atividades. Os alunos estavam habituados a trabalhar em pares, assim para a escolha das equipas, agrupou-se pares de alunos que trabalhavam bem mutuamente, e tendo as notas escolares como critérios juntamente com a professora cooperante, optou-se pela elaboração de equipas heterogêneas, resultando em duas equipas de três elementos e quatro equipas de quatro elementos.

5. Narrativas da Prática de Ensino Supervisionada

Vivemos através de histórias, pensamos, imaginamos e fazemos escolhas de acordo com estruturas narrativas (Oliveira, Segurado & Ponte, 1999, citado por Galvão, 2005). Organizamos a nossa experiência diária e a nossa experiência de acontecimentos principalmente sob a forma de narrativa. Criarmos histórias, desculpas, mitos, razões para fazer ou não fazer, é algo que nos é implícito e surge com naturalidade (Bruner, 1991, citado por Galvão, 2005). As narrativas são histórias que se reportam a acontecimentos passados e específicos e têm propriedades comuns (Labov, 1972, 1982, citado por Galvão, 2005). Uma narrativa é a apresentação simbólica de uma sequência de acontecimentos ligados entre si por determinado assunto e relacionados pelo tempo (Scholes, 1981, citado por Galvão, 2005).

Uma das vantagens do uso de investigação narrativa é que as histórias tornaram-se um meio de capturar a complexidade, a especificidade e inter-relação dos fenómenos com que lidamos (Carter, 1993, citado por Galvão, 2005). Cortazzi (1993) afirma que os métodos narrativos permitem ajudar os que estão fora das salas de aula a entender melhor o que acontece nas salas de aula do ponto de vista do professor. Isto porque as narrativas descrevem como os próprios professores veem determinada situação, como a experienciam, o que acreditam e o que pensam (Cortazzi, 1993). Para além de que, o próprio facto de escrever sobre a própria prática leva o professor a aprender através da narração (Cortazzi, 1993). Ao narrar a sua experiência recente, o professor constrói linguisticamente a experiência vivida, relembrando o discurso prático da experiência vivida segundo a sua perceção e repensa a sua atividade profissional, tornando a narração num processo de reflexão (Cortazzi, 1993). Este processo de reflexão envolve dar sentido, contar, transcrever, analisar e ler situações de contexto escolar (Cortazzi, 1993). A narrativa apresentada neste capítulo foi estruturada segundo o modelo de Labov composto por: resumo, orientação, compilação, avaliação e resultado (1972, 1982, citado por Galvão, 2005). Assim, as narrativas foram organizadas por aula, e em cada uma é feito um resumo da planificação prevista e objetivos pedagógicos; seguindo-se de uma orientação de quando decorreu, quem são os envolvidos e os recursos necessários; uma compilação das ações mais relevantes no decorrer da aula; uma avaliação das mesmas segundo o ponto de visto do professor; a respetiva resolução tomada e os resultados.

5.1. Concretização das aulas

A prática de ensino supervisionada decorreu na Escola Secundária de Camões no letivo 2012/1013 à disciplina de 12º ano Aplicações Informáticas B, no início do segundo período e ao longo de cinco aulas de noventa minutos o correspondente a três semanas. Os conteúdos abordados foram conceitos de programação e algoritmos, estes conceitos eram introduzidos através de reflexões iniciais, explorados através de desafios e refletidos em conjunto nas reflexões finais.

5.1.1. Narrativa da primeira aula.

Segundo o plano de atividades para a primeira aula (Anexo E), estavam previstas as seguintes atividades: organização das equipas; breve apresentação do professor; aplicação de um pré questionário de cariz investigativo; reflexão inicial de conceitos de programação; experimentação do Desafio 1 parte 1 com os objetivos de aprendizagem Desenvolver programas no robô através do i-Brick e Interpretar o comportamento de robô; e por fim a recolha o desafio. Para este desafio não foi preparado nenhum cenário específico, estando ao dispor dos alunos toda a área da sala de aula para experimentação da programação do robô, podendo os próprios livremente explorar e improvisar os seus cenários com os elementos existentes na sala de aula, desde mesas, cadeiras, estojos, cadernos, entre outros. Na sala de aula foram reservados computadores especificados para equipa e afixados no quadro branco papeis com a constituição das equipas. Os seis robôs correspondentes às seis equipas já haviam sido montados. Estavam disponíveis dois Spikes, dois Tribtos e dois Basquetebolistas. A todos estes modelos de robôs foram aplicados os sensores: ultrassónico, som, toque e luz/cor.

No primeiro dia de intervenção letiva, assim que os alunos chegavam à sala de aula, encaminhava-os para o quadro branco onde estavam afixados cartões com a constituição das equipas e indicava que deveriam sentar-se nos lugares perto dos computadores com os números correspondentes ao número da sua equipa. Assim que os alunos se sentaram nos respetivos lugares, apresentei-me e falei brevemente sobre os objetivos da intervenção letiva. Seguidamente, informei os alunos que estava disponível do *Moodle* da escola uma nova disciplina chamada NXT Heroes, na qual se encontrava uma hiperligação para um questionário de preenchimento online e individual (Anexo S). Os alunos demoraram mais do que o tempo previsto de 15 minutos para o preenchimento

do questionário, assim sendo teria que ao longo da aula ir fazendo alguns ajustes nos tempos previstos, de modo a não dar mais ênfase às atividades mais teóricas do que às práticas. Seguiu-se a Reflexão inicial sobre conceitos de programação (Anexo Z) e a apresentação da temática dos desafios pelo visionamento de um vídeo sobre a história NXT Heroes onde vilões decidem roubar todos os doces e *fast-food* de uma cidade. Um dos super-heróis que protegia a cidade ao tentar impedi-los é capturado. O outro super-herói, a Mulher Invisível tenta preparar um conjunto de alunos para salvar a cidade através da programação de robôs. Para isso, leva-os para um lugar secreto e dá início à Semana de Treino. A relutância dos alunos em relação à história tornou-se evidente através de comentários, tais como, “Que idade tinham os alunos que criaram esta história?” “Mas o Wolverine e o IronMan na realidade não são vilões”, “Esta história não faz muito sentido”. Ponderei com os alunos que apenas por uma questão de falta de tempo não lhes tinha dado oportunidade de elaborar a sua própria história e que apesar que não concordarem totalmente com aquela história ela era apenas o início da aventura e que a partir dali eles iriam viver e retratar todo o desenrolar da mesma. Com os alunos mais conformados fizemos em conjunto um resumo da história do vídeo de modo a perceberem o contexto do desafio.

Seguidamente afirmei que estava disponível no *Moodle* para *download* a proposta do Desafio1. Os alunos espontaneamente viraram-se para os computadores e acedem rapidamente ao ficheiro, perdendo assim a sua atenção. Havia previsto a explicação do recurso NXT Heroes criado para o Desafio1, mas encarrei o ato dos alunos como uma necessidade de atividades práticas. Para além do que, o recurso era bastante explícito em relação às tarefas que os alunos deveriam realizar e continha indicações de como fazê-las. Considerando que tinha estado com os alunos sensivelmente 40 minutos em atividades reflexivas, decidi saltar essa parte e permitir que os alunos explorassem o recurso por si mesmos. Enquanto o faziam pedi que elessem um elemento de cada equipa para escolher um envelope e recolher o robô especificado nele. A equipa 1 ficou com o envelope C, a equipa 2 ficou com o envelope A, a equipa 3 ficou com o envelope E, a equipa 4 ficou com o envelope F, a equipa 5 ficou com o envelope D e a equipa 6 ficou com o envelope B.

Não esperei muito tempo e iniciei o acompanhamento das equipas, começando pela equipa 1 consecutivamente até à 6. Nesse momento, os alunos estavam a trabalhar individualmente para os objetivos pedagógicos Desenvolver programas no robô através

do i-Brick e Interpretar o comportamento de robô. Passado sensivelmente cinco minutos do início desta atividade, já se ouviam robôs em movimento. Deparei-me com o facto de quando me aproximava os alunos perguntavam-me o que era para fazer e quando os questionava sobre o que o Desafio 1 dizia para fazer, estes não sabiam responder. Apercebi-me que os alunos não estavam a ler todos os slides do recurso NXT Heroes, o que fazia com que não conseguissem distribuir tarefas, nem trabalhar com o robô de acordo com os objetivos de aprendizagem previamente definidos. Optei por em cada equipa ajudar os alunos na divisão de tarefas, identificando aquele que iria programar em primeiro lugar, quem iria ser o seu ajudante, salientando que todos deveriam ler com atenção o slide 3 do Anexo P e depois o programador e ajudante começariam a trabalhar com o robô, enquanto os restantes elementos da equipa iniciariam o preenchimento dos requisitos do recurso NXT Heroes, e que depois rodassem de papéis. À medida que ia fazendo o acompanhamento das equipas, a relação aluno-professor foi-se fortalecendo e os alunos de forma voluntária começaram a ir ao meu encontro, com a folha de testes no i-Brick (Anexo R) em mãos, para esclarecimento de dúvidas.

Os alunos ao programar diretamente no i-Brick não sentiram dificuldades a nível de sintaxe, dando tempo para que explorassem os comandos disponíveis e através de testes interpretassem o comportamento do robô. Esta interpretação dava sentido aos programas criados e ajudava os alunos a pensar e melhorar os seus programas. Relativamente à interação com o robô, quando se programa no i-Brick e têm-se todos os sensores ligados, ao correr o programa o sensor de luz acende uma luz de presença vermelha. Este facto induziu em erro algumas equipas uma vez que apesar de não o usarem diziam que sim, por pensarem que essa luz era resultado do seu programa, mesmo que isso não fizesse muito sentido. Por outro lado, aqueles alunos que usavam o sensor de luz devido à luz de presença consideravam que era esse o efeito esperado e não que o robô estivesse à espera de receber um *input* de “light” ou “dark”. Nestes casos, perguntava às equipas em que momento do programa estavam à espera de uma ação do sensor de luz e refletia com elas que a luz de presença acendia desde o início do programa, logo não poderia ser esse comportamento programado. Depois dizia para interagirem com o sensor e ver o que acontecia. Em outra circunstância, uma aluna questionou-me “Professora o meu Robô deve estar avariado porque não está a fazer som”. Ao que lhe perguntei se ela tinha programado o robô para emitir ou receber um som. A aluna sorriu e foi para o seu lugar e ao testar novamente o robô, chega à conclusão que o

sensor de som reage quando deteta barulho. Um outro aluno considerava que o sensor ultrassónico do robô via os objetos ao embater contra eles e então mudava de direção. Refleti com o aluno que cada sensor representava figurativamente um sentido humano e usando as suas palavras disse para testar o robô mais uma vez e discernir se o robô de facto via ou sentia os objetos. Com esta pergunta em mente o aluno ao fazer testes no robô chegou à resposta certa, de que o sensor ultrassónico pode ser programado para detetar obstáculos até determinada distância. Apesar disso o aluno decidiu não registar a sua solução, afirmando que queria testar uma solução diferente. Salientei que por uma questão de tempo ele deveria ficar com aquela solução e fazer o registo do comportamento do robô, sendo que os restantes elementos da equipa precisavam do robô. No entanto, o aluno não seguiu a minha orientação e no final da aula entregou uma solução diferente. Aproximando-se o terminar da aula pedi aos alunos que me entregassem as folhas de Teste no i-Brick, que guardassem o recurso NXT Heroes de cada equipa na Pen Drive e que desligassem os computadores.

Em casa ao refletir sobre o decorrer da aula, considerei positivo o interesse dos alunos pelos programas das outras equipas, observando-as atentamente enquanto testavam os programas. Considerei que as minhas perguntas reflexivas mais usadas: O estavas à espera que aconteça? E o que fez o robô? Tinham surtido efeito positivo nos alunos. Ao fazer a avaliação formativa do desempenho dos alunos, constatei que de 22 alunos naquela aula 17 tinham conseguido apresentar uma solução. Alertou-me o facto que numa das equipas com quatro elementos apenas me apresentarem uma solução, de modo que coloquei nas minhas recomendações para a aula seguinte estar atenta ao ritmo de trabalho dessa equipa. Elaborei um feedback escrito para cada um dos 17 alunos sobre como poderiam melhorar as suas soluções (Anexo T). Constatando que a maior parte dos erros dos alunos denotavam falta de atenção à leitura do recurso NXT Heroes, mais especificamente ao slide 3 do anexo P, onde constavam as regras, sendo que alguns alunos haviam feito uso de empty's, e outros não usaram sensores na sua solução, mas na sua maioria o feedback referia para usarem mais sensores nas suas soluções e para dentro da equipa explorarem sensores diferentes. Senti a necessidade de ajudar os alunos a organizar o seu trabalho e por isso criei uma tabela com a ordem de programadores para a aula seguinte e disponibilize-a no *Moodle*. Deste modo quis certificar-me de que quem não havia programado iria assumir esse papel na aula seguinte e que os alunos usariam o restante tempo para melhorar os seus programas. Nas minhas notas do decorrer da aula,

tinha um apontamento para rever o nível de dificuldade do Desafio 1. Face à avaliação feita considerei que os alunos estavam no bom caminho, o que indicava que o nível de dificuldade estava apropriado.

5.1.2. Narrativa da segunda aula.

No plano de atividades para a segunda aula (Anexo E) estava prevista depois da organização das equipas, a continuação da experimentação do Desafio 1 parte 1 de modo a todos conseguirem programar o robô criando o seu programa individual, com os objetivos de aprendizagem: Desenvolver programas no robô através do i-Brick e Interpretar o comportamento de robô. Seguindo-se a experimentação do Desafio 1 parte 2 com o objetivo de aprendizagem Compreender conceitos de programação, que incluía: Definir o conceito de programação, Identificar o conceito de programação, Listar os comandos disponíveis no i-Brick, Identificar sensores do robô e suas funcionalidades e Elaborar uma história No final da aula estava previsto um momento de reflexão final do desafio em si. Findo o tempo de exploração do Desafio 1 parte 2, a cada equipa recriaria para a turma a história do programa de equipa e depois em conjunto seria feita uma reflexão final sobre os conceitos de programação aprendidos com o desafio.

Iniciei a aula por encaminhar os alunos para os computadores reservados para cada equipa. Salientei que estavam a desenvolver um bom trabalho mas que havia ainda aspetos a melhorar e lembrei-os que a parte inicial da aula estava reservada para a continuação da experimentação do Desafio 1 parte 1. Junto aos computadores iriam encontrar uma folha amarela impressa com feedback escrito individual dos programas elaborados na aula anterior. Mas de modo a todos terem oportunidade de programar o robô, deveriam respeitar a listagem de programadores que havia disponibilizado no *Moodle*. Assim, os alunos deram início à continuação do Desafio 1 parte 1, com os objetivos pedagógicos: Desenvolver programas no robô através do i-Brick e Interpretar o comportamento de robô. Existia muita atividade na sala de aula. Reparei que as equipas estavam a trabalhar em conjunto. Apesar de os programas serem individuais, todos os elementos da equipa acompanhavam o programador e davam sugestões sobre a interpretação do comportamento do robô, era uma aprendizagem coletiva entre pares. Os alunos revelaram-se curiosos em relação ao feedback escrito, colocando questões, mostrando-se ansiosos por voltar a programar o robô e melhorar os seus programas.

Apercebi-me que os vinte minutos que tinha previsto para melhoramento do Desafio 1 parte 1 não eram suficientes para que todos tivessem uma segunda oportunidade de melhorar o seu programa. Logo decidi dar mais 10 minutos para esta atividade, sendo que os alunos estavam envolvidos no desafio e queriam aprender ainda mais com ele.

Em relação à programação no i-Brick, os alunos que estavam a trabalhar com o sensor de som estavam com dificuldades. Devido à dinâmica existente na sala de aula, existia barulho salutar na sala de aula, resultante de alunos a movimentarem-se pela sala a testar o seu robô e a equipas a argumentar sobre a melhor interpretação do comportamento do robô, fazendo com que o robô reagisse espontaneamente ao som, sem que fosse um som provocado pelo seu programador, o que confundia a interpretação do comportamento de robô. A professora cooperante sugeriu que estes alunos testassem os seus programas no corredor fora da sala de aula e assim os alunos conseguiram o ambiente necessário para testar os seus programas, conseguindo acaba-los e interpretá-los corretamente. Interessante o facto de que em momento algum os alunos me pediram tarefas extra, nem me disseram que haviam terminado o Desafio parte 1, apesar de eu me aperceber que algumas equipas já haviam terminado. A verdade é que mesmo terminando o Desafio 1 parte 1 os alunos de livre vontade continuavam a explorar a programação no robô, completando o seu conhecimento. Durante estes 30 minutos os robôs nunca pararam.

Enquanto realizava o acompanhamento das equipas, os alunos faziam muitas questões e notei que os alunos já não usavam expressão tais como “o robô está avariado”, “o robô não funciona”, ou “não percebo o robô”, usados com frequência na aula anterior. Nesta aula os alunos quando me procuravam por ajuda, exponham o concretamente qual a sua dificuldade, evidenciando que estavam a refletir em onde havia errado e não reportando o problema para o robô em si. Notei que a equipa 3 apesar de me dizer que não tinha dúvidas e que não precisava de ajuda, não avançava na programação do robô. Questionei a equipa sobre este facto, ao que o programador reconheceu que afinal precisava de ajuda. Perguntei-lhe porque não me havia dito nada das outras vezes, ao que respondeu que tinha vergonha de mostrar que não sabia. Reconfortei o aluno por dizer que todos estavam ali para aprender e que os seus colegas também tinham dúvidas, era algo natural e que eu estava à espera que acontecesse, e que eu estava ali para ajudar. Reafirmei que os alunos não perdiam pontos por fazer perguntas, pelo contrário que isso mostrava que estavam interessados, e contava positivamente para a avaliação das atitudes

e comportamentos. Ajudei o aluno a pensar o seu problema, este não percebia o porquê de o robô parar, ao que expliquei que o robô estava à espera de algo, o aluno associou logo o algo ao sensor, deu o *input* necessário ao robô e solucionou o seu problema. Ao ir embora disse ao aluno “Vês, tu sabias a resposta apenas precisavas de uma pequena ajuda e é para isso que estou aqui. Podias-me ter chamado mais cedo, não achas?”, ao que o aluno disse que sim em resposta. Notei que esta equipa era a que menos interagiu entre si, facto evidenciado pelos colegas não ajudarem este aluno na resolução do problema e também não estarem interessados na minha explicação.

Terminados os 30 minutos os alunos foram alertados que deveriam passar para a parte 2 do Desafio 1, correspondente aos slides 4, 5 e 6 do recurso NXT Heroes. Nesta parte do desafio os alunos seleccionando um dos programas desenvolvido por um dos elementos da sua equipa, em equipa teriam que evidenciar o cumprimento dos objetivos de aprendizagem Compreender conceitos de programação por Definir o conceito de programação, Identificar o conceito de programação no programa selecionado; Listar os comandos disponíveis na programação direta no i-Brick; Identificar dos sensores do robô que tinham sido utilizados no programa selecionado e suas funcionalidades; e por Elaborar uma história interpretativa e contextualizadora do comportamento do robô. Maioritariamente as equipas organizaram o seu trabalho segundo a sequência dos slides do recurso NXT Heroes, tendo resolvido com alguma rapidez o slide 4 relacionado com a listagem de comandos do i-Brick. No entanto, as equipas demoraram muito do seu tempo no slide 5 correspondente à elaboração da história, descurando o slide 6 que era o muito importante. Face a esta situação, tentei que os alunos repartissem tarefas, por exemplo nas equipas de quatro elementos aconselhei que dois se encarregassem pela história enquanto os outros dois adiantavam o slide 6. Senti os alunos muito inseguros em relação ao slide 6, porque teriam que refletir sobre a sua experiência de aprendizagem e transpor isso em texto. Os alunos perguntavam-me muitas vezes se estavam a preencher corretamente o slide. Face a esta insegurança e alguma dificuldade inicial, decidi abdicar na reflexão final de conceitos de programação, uma vez que os alunos demonstravam que não estavam preparados para esse momento e esse tempo foi usado para apoiar individualmente cada equipa na sua reflexão. Mas antes de iniciar o acompanhamento das equipas, disse para as equipas pararem por uns minutos e expliquei item a item para toda a turma o que era esperado em cada um e esclareci algumas dúvidas existentes quando questionada.

Aproximando-se o terminar da aula disse para os alunos para guardarem os ficheiros porque estava na hora da encenação das histórias. Os alunos continuavam a pedir tempo, mas finalmente acederam. Cada equipa teve oportunidade de encenar a sua história perante a turma. Enquanto um aluno da equipa interagiu com o robô, outro contava a história. Este revelou-se um momento agradável, com muitos sorrisos e aplausos. Feitas as apresentações, apesar de não faltar muito tempo para a final da aulas as equipas pediram o restante do tempo para melhorarem o slide 6. A equipa 4 não quis melhorar, mostrou-se confiante com o seu trabalho. Perguntei-lhes se tinham a certeza, ao que responderam que sim. No final da aula todas as equipas guardaram os seus trabalhos na Pen Drive.

Em casa ao realizar a avaliação das equipas, reparei que a equipa 4 apesar de parecer uma boa equipa, teve negativa na componente de reflexão de conceitos de programação. Ao refletir sobre a aula passada, lembrei-me que esta havia sido a equipa que não quis rever esta componente de avaliação. Esta situação fez-me considerar pertinente promover na aula seguinte um momento de auto e hetero avaliação do Desafio 1, de modo a que cada equipa tomasse conhecimento da minha perspetiva sobre eles, tendo a nível de atitudes e comportamentos, capacidades e competências. Alertando a equipa 4 para o seu excesso de confiança, a equipa 3 para me chamar quando tivessem dúvidas e comunicassem abertamente no interior da equipa. Considerei também importante alertar as equipas 5 e 6 para que melhorassem o seu comportamento e para que as equipas 1 e 2 interagissem mais nas reflexões de turma. Salienta-se que apesar de os alunos ainda terem aspetos a melhorar, todas as equipas haviam obtido resultados positivos, tendo os alunos no geral obtido resultados bastante satisfatórios. Durante a auto e hetero avaliação era esperado que cada equipa refletisse sobre o seu desempenho, depois eu revelaria a minha opinião, depois em conjunto refletíssemos sobre a concordância ou discordância entre as nossas perspetivas e pensássemos em soluções para as mesmas. No entanto, senti a necessidade de dar voz aos alunos não só em relação ao seu desempenho, mas em relação ao Desafio 1 em si e à forma como foi aplicado e uma vez que durante a aula não por falta de tempo e eu também não havia exigido, os alunos não chegaram ao slide 7 do recurso NXT Heores (Anexo P), elaborei um questionário extra (Anexo U) em substituição deste.

5.1.3. Narrativa da terceira aula.

No plano de atividades para a terceira aula (Anexo P) estava previsto o preenchimento de um questionário sobre o Desafio 1, a realização de uma auto e hétero avaliação do Desafio 1, a realização de uma Reflexão inicial sobre conceitos de algoritmos e a experimentação do Desafio 2. Segundo o tema do desafio, a Mulher Invisível mostra aos alunos uma ilha e diz que eles têm que chegar ao castelo no topo, mas para terem cuidado que este era vigiado pelo terrível Lama. Para a concretização deste desafio foi construído um cenário (Figura 17), onde a rampa prefigura o túnel, mas mesas a ilha secreta e as caixas de cartão azuis as muralhas do castelo.

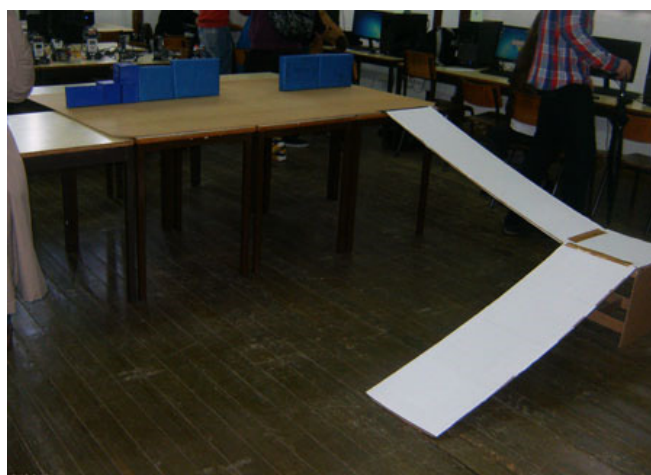


Figura 17. Cenário do Desafio 2

A aula iniciou-se pelo preenchimento de um questionário por equipa sobre o Desafio 1 (Anexo U). Visto os alunos na primeira aula terem demorado muito tempo no preenchimento de um questionário, este por se tratar de um questionário extra foi planeado para ser preenchido em equipa, de modo a não prejudicar os alunos a nível de tempo para execução dos desafios. Seguiu-se a auto e hétero avaliação referente ao Desafio (Anexo Z) e reflexão inicial sobre conceitos de programação e algoritmos.

Terminada reflexão, disse aos alunos que estávamos a iniciar o Desafio 2, e que agora iriam programar o robô de forma diferente, em NXT-G. Para este desafio os objetivos de aprendizagem definidos eram: Desenvolver programas através do NXT-G aplicando o bloco MOVE e Elaborar pseudocódigo. De modo a pensar na estratégia para a resolução do desafio pedia a um aluno que se aproximasse do cenário e com a ajuda da turma percorre-se com o seu robô na mão o cenário de modo a conseguir chegar ao

castelo. A turma concordou que o desafio passaria por três fases distintas: o subir a rampa, atravessar a muralha e imobilizar o robô. Sem entrar em mais detalhes disse aos alunos para se fixarem na primeira fase, subir a rampa. Salientei que cada equipa controlaria o robô de uma forma diferente, e que deveriam seguir a orientação do papel laranja que estava no envelope. Existindo assim as possibilidades: rotações, graus, tempo e ilimitado. Sendo que agora controlariam de forma diferente o robô, dei a dica aos alunos que deveriam descobrir qual era o bloco que fazia o robô andar. Ouvi alguns alunos a questionarem-se sobre o que era um bloco, mas não me expressei, porque sabia que ao experimentarem por eles o NXT-G, chegariam logo à conclusão. Salientei igualmente que precisariam do cabo USB para fazer a comunicação entre robô e o NXT-G.

Passado algum tempo os robôs já se mexiam. A equipa que ficou com a característica tempo não teve dificuldades. Quanto à equipa que ficou com a propriedade ilimitado, logo se apercebeu que não significava o robô andar indefinidamente, mas que quando associada a uma estrutura de repetição o robô andava sem soluçar, contrariamente às outras formas de controlar o robô. Esta equipa decidiu que iria controlar o robô através dos botões do i-Brick, fazendo o robô andar indefinidamente para a frente quando pressionado o botão cor de laranja, andar indefinidamente para a direita quando pressionada a seta da direita e andar indefinidamente para a esquerda quando pressionada a seta da esquerda. As duas equipas que ficaram com a característica rotações, quando lhes perguntava o que significava uma ou duas rotações não sabiam responder apesar de terem testado o robô, e conseguirem fazer o robô andar. Dizia aos alunos para associarem as rotações ao elemento do robô que o fazia mover, ao que os alunos chegaram à conclusão que eram as rodas, logo só poderia ser a rotação das rodas. A propriedade graus suscitou dúvidas entre os alunos, porque para eles graus 90, 180 ou 360° significaria o robô rodar em torno do seu próprio eixo esses ângulos específicos, só que ao testar o robô isso não acontecia. Quando questionava os alunos sobre o comportamento do robô, diziam que independente do ângulo inserido o robô andava em frente e não para os lados. Então dizia para os alunos pegarem no robô, o colocarem o dedo indicador sobre uma das rodas, e depois dizia “Se fosses o robô e te programassem para andar 90° o que fazias? E 180°? E agora 360°?” (Figura 18) Através da visualização, experimentação e simulação dos movimentos do robô, os alunos finalmente entendiam o que significava os graus.



Figura 18. Exemplificação de uma rotação no robô em graus

À medida que os alunos entendiam a sua propriedade de controlo da Duração do movimento do robô, passavam a compreender a utilidade da fita métrica, medindo o cenário e efetuando cálculos matemáticos para obter maior precisão no movimento. Uma das equipas estava a testar o robô a subir a rampa e dizia: “Professora é difícil estar a contar quantas rotações o robô faz ao subir a rampa ele é muito rápido”. Fazendo-lhe referência para o slide 10, perguntei quantos centímetros andava o robô em uma rotação. Ao que os alunos responderam que ainda não haviam calculado, mas que já percebiam que ao media a rampa e fazer as conversões era mais eficaz obter o número de rotações para controlar o robô. Notei que a equipa que controlava o robô através do tempo mediam quanto andava o robô em um segundo, criaram uma medida de referência e contavam quantas vezes tinham que repeti-la.

Os alunos que ficaram com o basquetebolista tinham uma dificuldade acrescida, o facto do robô ao subir a rampa cair para trás. De modo aos alunos perceberem a multidisciplinaridade da programação com robôs, as equipas tinham à sua disposição, sacos com pedras, cordas, fita-cola e peças lego. No entanto, os alunos não consideraram este facto como uma oportunidade de aprendizagem, mostrando-se relutantes em mexer na estrutura do robô. Ver o robô das outras equipas subir a rampa sem cair para trás, deixava-os desanimados e pediam-me persistentemente que fizesse as alterações necessárias. Uma das equipas não se mostrou disponível para negociação e foi necessário prover-lhes um robô alternativo, enquanto a outra equipa continuou com o robô descontente e em constante queixa e lamúria, dizendo que o robô deles não prestava. No final da aula uma das equipas conseguiu subir a rampa e turma reagiu batendo palmas.

5.1.4. Narrativa da quarta aula.

No plano de atividades para a quarta aula (Anexo E) estava prevista a conclusão do Desafio 2 e realização de uma reflexão final sobre conceitos de programação e algoritmos. No entanto, numa breve análise ao questionário do sobre o Desafio 1, reparei nos comentários dos alunos, estes afirmaram precisar de mais tempo para programar o robô (Anexo U). Tendo em conta a opinião dos alunos decidi dar mais tempo aos alunos. Assim, a quarta aula seria dedicada para a experimentação do Desafio 2 passando a reflexão final sobre conceitos de programação e algoritmos para a quinta aula.

A aula iniciou-se por relembrar o contexto de aprendizagem aos alunos, salientando que estavam ainda na Semana de Treino, sendo que os vilões poderiam neutralizar os sensores dos robôs, os alunos deveriam continuar a controlar os robôs sem o uso de sensores.

Durante o decorrer da aula os alunos não revelaram dúvidas ter muitas dúvidas a nível de programação em NXT-G parecendo ter-se adaptado bem a este tipo de programação. A maior parte dos obstáculos eram relacionados com outros aspetos. Como por exemplo, os alunos não iniciavam o robô sempre no mesmo lugar do cenário, o que fazia com que para o mesmo programa, às vezes o robô conseguissem cumprir o seu objetivo e outras vezes não. Para as equipas que estavam a usar o robô basquetebolista, acrescentei uma base para que apenas tivessem de pousar a saco de pedras para equilibrar o peso do robô. Este facto minimizou as queixas em relação ao robô.

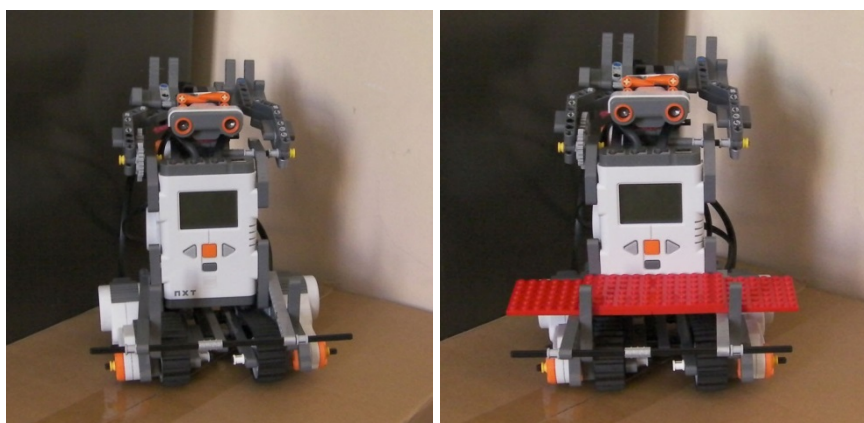


Figura 19. Robô basquetebolista alterado

Outras equipas estavam a demorar na concretização do Desafio 2 porque tinham deixado a usar a fita métrica para calcular com mais exatidão a Duração do movimento do

robô. Nesses casos, sugeria para voltarem a recorrer à fita métrica para medição das distâncias e com o auxílio do slide 10 do Anexo P, e acompanhava as equipes neste processo. Estas ao ver que resultava voltavam a usá-la. Em relação a outros obstáculos nesta aula um dos alunos na equipe 1 que a equipe estava a ter divergência de ideias e que estas estavam a afetar o desempenho da equipe. Face esta situação sugeri que para cada fase do Desafio 2 definissem previamente um programador e ajudante, e em cada programa o programador teria a palavra final, mas que este apesar de ter a palavra final não significava que devessem ignorar completamente a opinião dos colegas. Nesta aula tive que intervir esta equipe várias vezes como mediadora, fazendo com que a equipe primeiro ouvisse a opinião de todos os seus elementos e depois tomasse decisões assertivas testando o robô.

5.1.5. Narrativa da quinta aula.

No plano de atividades para a quinta aula estava previsto um momento de reflexão sobre a intervenção em si, mas como ainda não havia sido feita a reflexão final sobre o conceitos de programação e algoritmos programada para quarta aula, decidi fazer uma reflexão sobre o Desafio 2.

Em relação ao controlo do movimento do robô, perguntei se todas as equipes tinham conseguido subir a rampa. Ao que as equipes responderam que sim. Com a finalidade de os fazer entender as vantagens de utilizar diferentes tipos de Duração para controlar o movimento do robô, questionei às equipes que haviam usado rotações, quantas haviam usado para subir a rampa. Ao que responderam entre quatro e cinco rotações. A seguir perguntei às equipes que haviam usado graus, quantos haviam usado. Uma das equipes respondeu 3300 graus. A turma mostrou-se surpreendida com o valor, e concluíram que para subir a rampa através de rotações seria muito mas simples a nível de cálculos. Depois reporte os alunos para a curva da rampa e a situação inverteu-se. As equipes que usaram rotações referiam que tiveram que trabalhar com valores com casas decimais entre 1 e 0 para fazer a curva, o que não era tão fácil acertar na precisão da curva. Nestes caso as equipes de usaram graus tinham uma maior precisão no controlo na rotação das rodas, através de valores entre 0 a 360°. Salientei que em provas de pistas de obstáculos sem sensores, as forma de controlar o robô mais usados são graus e rotações. Mas não esquecendo que poderiam usar o tempo. E que em exercícios com sensores é

mais usual usar a propriedade ilimitado. No entanto, todos haviam comprovado que era possível concluir a desafio controlando o robô com qualquer tipo de Duração do movimento do robô.

Em relação ao robô Basquetebolista perguntei às equipas que o usaram como solucionaram o facto de o robô cair para trás ao subir a rampa. Responderam por colocar mais peso no robô através do saco com pedras. Perguntei que disciplina estava presente nesse ato para além da programação e da robótica. Os alunos referiram a física. Direcionando-me para as outras equipas, perguntei que disciplina extra tinham aplicado durante o Desafio 2. Ao que responderam a matemática, ao efetuar as conversões de centímetros para rotações, graus ou tempo. Salientei à turma que a programação era uma disciplina multidisciplinar, tal como eles tiveram oportunidade de comprovar. Depois desta reflexão foi dado tempo para os alunos melhorarem o Desafio 2 e pedido que preenchessem dois questionários (Anexo V, W).

5.2. Avaliação da Prática de Ensino Supervisionada

No plano de avaliação da prática de ensino supervisionado a avaliação final dos alunos era composta 90% por Conhecimentos Competências e Capacidades e 10% pelas Atitudes e Comportamentos. A nível de atitudes e comportamento, esta avaliação foi elaborada conjuntamente com a professora cooperante (Anexo X). Em relação aos Conhecimentos, Competências e Capacidades, agora constituída apenas pelo Desafio 1 (50%) e Desafio 2 (50%), visto a atividade extra ter sido retirada da avaliação por ter sido uma atividade que requereu muito apoio do professor, tal como referido na narrativa da terceira aula.

5.2.1. Avaliação do Desafio 1.

O Desafio 1 era composto por três parâmetros de avaliação: programação, interpretação e reflexão. Tanto o parâmetro programação como interpretação foram desenvolvidos individualmente pelos alunos e valiam 50% da classificação do desafio, 25% cada. Enquanto a componente de reflexão foi realizada em equipa e valia os restantes 50% de classificação. Através do Quadro 10 podemos averiguar que a avaliação sumativa do Desafio 1 resultou em cem por cento de positivas, em que 14% das classificações foram suficiente, 50% bom e 36% muito bom.

Quadro 10. Classificações finais do Desafio 1

	Muito fraco	Insuficiente	Suficiente	Bom	Muito Bom
Classificação	0%	0%	14% (3)	50% (11)	36% (8)

Nota. Muito fraco 0 a 4.4; Insuficiente 4.5 a 9.4; Suficiente 9.5 a 13.4; Bom 13.5 a 17.4; Muito bom 17.5 a 20.

Analisando em detalhe as classificações do Desafio 1 por parâmetro, segundo o Quadro 11 a média de classificações mais alta foi a interpretação com 88%, seguindo-se a programação com 87% e por último a reflexão com 70%. Através do Anexo X pode-se observar com mais detalhe que houve cem por cento positivas à componente programação, com classificações compreendidas entre os 75% e 95%. Enquanto no parâmetro interpretação houve igualmente cem por cento de positivas, mas com classificações compreendidos entre 60% e 100%. Por último no parâmetro reflexão existiu uma classificação negativa, que sendo de trabalho em equipa foi atribuída por igual aos três elementos que a constituíam. Salienta-se que na narrativa da segunda aula foi descrito que a equipa 4 ao contrário das restantes, quando dada a oportunidade de melhorar a reflexão não o quis fazer. As equipas 1 e 2 obtiveram as classificações mais altas no Desafio 1 e foram igualmente as equipas que obtiverem melhor classificação no parâmetro reflexão. Ou seja, evidenciaram ser equipas que para além de conseguirem programar e interpretar o comportamento do robô, mostraram capacidade de reflexão sobre os conceitos de programação envolvidos no trabalho que estavam a realizar. As outras equipas de modo geral conseguiram com sucesso programar e interpretar o comportamento do robô, mas a nível de classificação do parâmetro reflexão ficaram consideravelmente abaixo, o que denota que não conseguiram de uma forma tão profunda compreender os conceitos por detrás no trabalho que estavam a desenvolver. No geral todas as equipas conseguiram alcançar classificações positivas, indicando que todas obtiveram aprendizagem.

Quadro 11. Médias de classificações por parâmetro do Desafio 1

	Média
Programação	87%
Interpretação	88%
Reflexão	70%

Analisando com mais detalhes os componentes do parâmetro reflexão, através do Quadro 12 os alunos obtiveram melhor classificação na Identificação dos comandos do i-Brick com 100% seguindo-se da História com 82% e da Identificação dos sensores e sua atuação com 73%, da Definição de programa com 71%, da Aplicação Conceito de Programação com 60% e por último de Como se programa no i-Brick com 40%.

Quadro 12. Classificações do parâmetro Reflexão do Desafio 1

Definição de programa	Aplicação Conceito de Programação	Como se programa no i-Brick	Identificação dos comandos do i-Brick	Identificação dos sensores e sua atuação	História
71%	60%	40%	100%	73%	82%

Salienta-se o facto de que as equipas obtiveram classificações mais altas na Identificação dos comandos do i-Brick e mais baixas na descrição de Como se programa no i-Brick, apesar de ambas estarem relacionadas. No entanto para descrever os comandos do i-Brick as equipas teriam recorrer ao robô, mais especificamente explorar todos os comandos disponíveis no Menu NXT Program, enquanto para a explicação de como se programava no i-Brick e como se chamava esse tipo de programação, as equipas não chegariam lá apenas pelo manuseamento do robô, teriam que ler o recurso NXT Heroes. Estes factos podem evidenciar que os alunos tal como descrito na narrativa da primeira aula, não estavam a ler com atenção o recurso, e que por outro lado os alunos foram mais produtivos em atividades que envolviam diretamente o robô, ou onde a resposta poderia ser obtida através do manuseamento do robô.

Outro facto interessante é que apesar de os alunos em média obterem melhor classificação na Definição de programa do que na Aplicações de conceitos de programação, fazendo uma comparação das trajetórias entre as duas classificações (Figura 21) denota-se que três são ascendentes, duas são descendentes e uma não revela alterações. O que indica que para a maioria das equipas aprenderam mais sobre o que era a programação ao programar, ou seja, através da aplicação do conceito recorrendo à experimentação do robô, do que propriamente através do conhecimento da definição de programação em si.

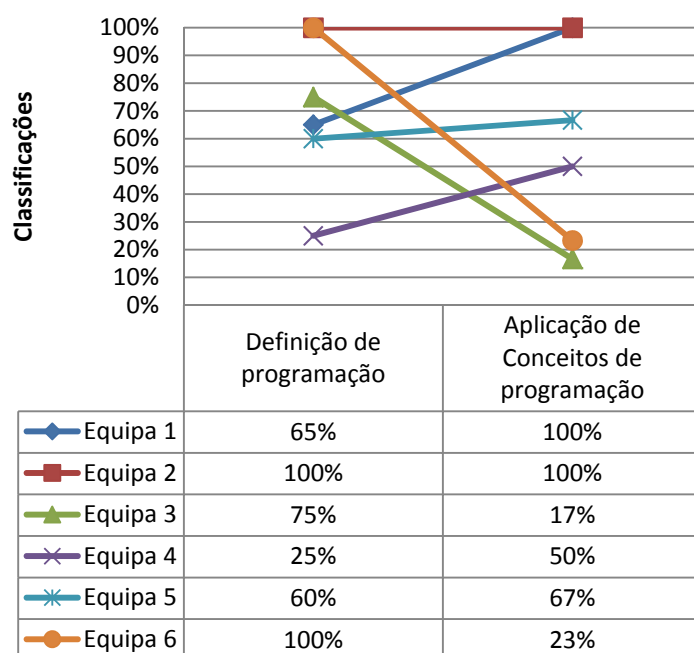


Figura 20. Análise de classificações entre Definir programação e Aplicação de Conceitos de programação

5.2.2. Avaliação do Desafio 2.

O Desafio 2 era composto por três fases, onde as equipes de alunos deveriam elaborar o programa, pseudocódigo e fazer o registo de vídeo. A fase um corresponderia ao subir da rampa (80%), a fase dois ao avançar da muralha (80%) e a fase três à habilidade especial do robô (20%). Apesar do aumento de complexidade e necessidade adaptação a uma nova forma de programar, neste caso NXT-G, do Desafio 1 para o Desafio 2 houve um melhoramento da média de classificações. Através do Quadro 13, pode-se averiguar que a avaliação sumativa do Desafio 2 resultou em cem por cento de positivas, sendo 73% das classificações bom e 27% muito bom.

Quadro 13. Classificações finais do Desafio 2

	Muito fraco	Insuficiente	Suficiente	Bom	Muito Bom
Classificação	0%	0%	0%	73% (16)	27% (6)

Nota. Muito fraco 0 a 4.4; Insuficiente 4.5 a 9.4; Suficiente 9.5 a 13.4; Bom 13.5 a 17.4; Muito bom 17.5 a 20.

Mais especificamente em relação ao parâmetro programação, através do Quadro 14, pode-se observar que este foi o parâmetro do Desafio 2 com melhores resultados com

média de classificação de 100%, traduzindo-se uma subida de 13% em relação ao Desafio 1. Salienta-se que quando os alunos entregavam o programa este revelava-se sempre atingir os objetivos de aprendizagem esperados. Este acontecimento pode dever-se ao facto do programa ter sido elaborado em equipa, ou até mesmo pelo gosto que os alunos desenvolveram em programar o robô no Desafio 1 associado ao maior controlo do robô proporcionada pela programação em NXT-G ter feito com que quisessem aprimorar a sua programação. No caso do parâmetro pseudocódigo a média de classificações para o Desafio 2 foi de 79%, menos 9% em relação ao Desafio 1. Possivelmente neste caso o aumento do grau de complexidade seja a razão da descidas da média de classificações.

Quadro 14. Comparação de médias de classificações por parâmetro

	Desafio 1	Desafio 2
Programação	87%	100%
Interpretação/Pseudocódigo	88%	79%

6. Cariz investigativo da prática de ensino supervisionada

A formação de professores segundo Nóvoa (2001) é algo que se estabelece num “continuum”, não é algo que se torna, é algo que se exerce. Entre os vários deveres da classe docente consta a “melhoria dos processos de ensino” visando um melhoramento “dos resultados escolares dos alunos” (Ministério da Educação e Ciência, 2012, p. 858). Surgem assim, denominações complementares à profissão docente, tais como professor investigador, pesquisador ou reflexivo, sendo estes professores aqueles que se preocupam com a qualidade do seu ensino e das suas práticas, agindo de acordo a melhorá-las através da pesquisa, reflexão e investigação (Nóvoa, 2001).

O tema deste relatório é “A Robótica Educativa no Ensino e Aprendizagem de Conceitos de Programação e Algoritmos” e em termos de investigação pretendeu-se analisar o impacto da robótica educativa na aprendizagem dos alunos. Vários autores defendem que a robótica educativa tem um impacto positivo na aprendizagem dos alunos, porém grande parte dos estudos não apresentam fortes evidências do mesmo, para além do que em cada estudo o impacto é medido através de variáveis diferentes (McGill, 2012). Neste caso, o impacto da robótica educativa foi medido através das variáveis: atitudes e perceções e atenção. No entanto, as circunstâncias decorrentes da prática de ensino supervisionada, limitaram o estudo em termos de tempo e de amostra, visto a duração da mesma ter sido de cinco aulas de noventa minutos e a amostra se restringir aos alunos que participaram da mesma. Por isso, este estudo apresenta-se meramente como de cariz investigativo, no sentido que regeu-se por procedimentos de investigação, contudo os resultados da análise não podem ser generalizados. Assim, não foram apresentadas conclusões da análise de dados, sendo no final do capítulo apenas apresentada uma breve discussão e considerações.

6.1 Problemática da dimensão investigava

No capítulo três Enquadramento curricular e didático da intervenção letiva abordaram-se os problemas no ensino e aprendizagem da programação, com especial ênfase nos dilemas: utilidade, sintaxe e prática (Bereiter & Ng, 1991, citados de Gomes et al., 2008; Gomes et al., 2012). De modo a fazer os alunos perceberem a utilidade nos programas produzidos, foram planificados contextos de aprendizagem, onde os alunos face a um desafio exploram formas próprias de o resolver através do robô (Rusk et al.,

2008). Em relação à sintaxe, para programar o robô foram escolhidos *softwares* de programação visuais: programação direta no i-Brick e NXT-G (Sebesta, 2003). Por último a nível de prática, a planificação das aulas num esforço de ir ao encontro das necessidades dos alunos em ter uma maior componente prática nas aulas em detrimento de teoria, garantiu que no mínimo por cada bloco de aulas de noventa minutos, quarenta e cinco fossem reservados para atividades práticas com robôs (Gomes et al., 2012). Como se pode constatar existe um denominador comum entre as soluções apresentadas para os dilemas da aprendizagem de programação, o robô.

No entanto, o robô para além de ser utilizado como um meio de minimizar os dilemas do ensino e aprendizagem de programação, também pode servir de estímulo para a aprendizagem dos alunos. Assim, achou-se pertinente estudar que impacto teve o robô na aprendizagem de programação, do ponto de vista dos alunos que experienciaram esta intervenção letiva. Se, tal como no estudo de Kumar e Meeden (1998) nesta intervenção letiva os robôs fascinaram os alunos e se esse fascínio fê-los querer aprender. Por outro lado, segundo os critérios de avaliação da disciplina de Aplicações Informáticas B na componente Atitudes e Comportamento um bom aluno seria aquele reúne as seguintes características: pontualidade, apresenta do material necessário, presta atenção nas aulas, possui sentido de oportunidade, espírito crítico, espírito de iniciativa, interage com o professor e a equipa, cumpre das tarefas propostas, é respeitador, é responsável, utiliza correta dos equipamentos e respeito pelas normas de funcionamento das salas. De todas estas qualidades salienta-se a atenção, sobre a qual vários estudos em sido realizados no sentido de comprovar o efeito positivo que a robótica tem em captar a atenção dos alunos. Partindo do pressuposto de que alunos atentos aprendem melhor, achou-se pertinente estudar o impacto que o robô teve sua na atenção.

6.2 Abordagem metodológica

A investigação de cariz descritivo visa observar, registrar, analisar e correlacionar fenômenos ou factos, sem interferir no ambiente analisado e sem manipulação de variáveis (Hill & Hill, 2012). Em todos os planos descritivos o objetivo na investigação é recolher dados que permitam descrever da melhor maneira possível comportamentos, atitudes, valores e situações (Hill & Hill, 2012). Neste caso específico trata-se de um estudo longitudinal, porque durante um período de tempo os mesmos sujeitos foram avaliados ou medidos mais do que uma vez.

6.2.1. Caraterização da amostra.

A prática de ensino supervisionada decorreu ao longo de cinco aulas de noventa minutos, numa turma de 12º ano de Aplicações Informáticas B, do curso científico-humanístico de ciências e tecnologias, composta por 22 alunos. Durante a intervenção letiva, a robótica educativa foi utilizada como estímulo para a aprendizagem de programação, mais especificamente aos conteúdos programáticos: conceitos de programação e algoritmos. Este foi o primeiro ano letivo que a disciplina de Aplicações Informáticas B entrou em funcionamento na Escola Secundária de Camões, mas não foi a primeira vez que se usou a robótica educativa em contexto curricular (ESC, 2012). Na disciplina extinta de Área de Projeto a robótica educativa era usada com sucesso e amplamente aceite pela comunidade escolar (ESC, 2011). Com intuito de reavivar a investigação sobre o uso da robótica educativa em contexto curricular, a disciplina de Aplicações Informáticas B foi planeada de modo a contemplar a robótica educativa na unidade de ensino-aprendizagem de Introdução à Programação. Antes da realização da prática de ensino letiva, sabia-se que os alunos atribuíam alguma importância à disciplina de Aplicações Informática B e à unidade de ensino-aprendizagem de Introdução à programação. Para além disso, os alunos revelaram ter algum interesse em experimentar a programação com robôs, havendo mesmo dois alunos que já haviam programado, mas não em contexto curricular e não a tecnologia que iríamos utilizar em sala de aula. Um desses alunos afirmou possuir um *kit* de Lego® Mindstorms® NXT apesar de nunca o ter experimentado.

Para a realização deste estudo foi aplicado um método de amostragem por quotas, procedendo-se a uma escolha de amostra não-aleatória de tamanho determinado pela fração de amostragem (Hill & Hill, 2012). Idealmente este estudo poderia ser realizado em mais turmas, ou até mesmo com um grupo de controlo, não havendo essa possibilidade, visto existir apenas uma turma de Aplicações Informáticas B na escola e esta não estar repartida por turnos, considerou-se a amostra total igual ao universo, ou seja, 22 participantes o correspondente ao número de alunos da turma de Aplicações Informáticas B. Devido à amostra de casos não ter sido escolhida aleatoriamente, não é possível extrapolar com confiança para o universo os resultados e conclusões tirados a partir desta amostra (Hill & Hill, 2012).

6.2.2. Variáveis em estudo.

O foco da investigação incide sobre o robô, pretendendo-se analisar o seu impacto no ensino e aprendizagem de programação à disciplina de Aplicações Informáticas B, na perspetiva dos participantes. Diversos estudos se têm feito sobre o impacto da robótica educativa na educação, mas tendo em consideração a brevidade da intervenção letiva de apenas cinco aulas de noventa minutos, escolha recaiu sob duas das suas variáveis analisadas no estudo de McGill (2012): atitudes e perceções, e atenção. A variável atitudes e perceções teve como finalidade medir as expectativas dos participantes face à programação e à programação com robôs. A fundamentação teórica usada por McGill (2012) para justificar a escolha da variável atenção, baseia-se no ARCS Model of Motivational Design desenvolvido por Keller (1987). Segundo o autor (1987) a atenção pode ser obtida de forma perceptiva ou por inquérito. A excitação perceptiva, usa de surpresa ou incerteza para ganhar interesse, por outro lado, a excitação por inquérito estimula a curiosidade através de questões desafiadoras ou problemas a serem resolvidos. Neste caso específico, toda a planificação da prática de ensino supervisionada centrou-se em prover contextos de aprendizagem estimulantes, que vão se encontro com a estimulação por inquérito da atenção, com especial ênfase nos momentos de reflexão. Mas em termos de investigação, este estudo pretende analisar do ponto de vista dos participantes se a robótica educativa, personificada pelo robô e aplicada no Desafio 1 e Desafio 2 proporcionou algum tipo de estímulo perceptivo da atenção e consequentemente serviu como impulsionador da aprendizagem, partindo do pressuposto que alunos mais atentos aprendem melhor.

6.2.3. Metodologias de recolha e análise de dados.

O método de recolha de dados utilizado foi inquérito por questionário, devido à sua maior rapidez e facilidade com que permite fazer a análise de dados (Quivy & Campenhoudt, 2008). O questionário é um instrumento de observação não participante, baseado numa sequência de questões escritas, que são dirigidas a um conjunto de indivíduos, envolvendo as suas opiniões, representações, crenças e informações factuais, sobre eles próprios e o seu meio (Quivy & Campenhoudt, 2008).

Para medir o impacto da robótica educativa no processo de aprendizagem dos alunos, segundo as variáveis atitudes e perceções, e atenção foi utilizado o instrumento

elaborado por MacGill (2012) baseado no instrumento validado de Keller (1987b) Instructional Materials Motivation Survey [IMMS], que por sua vez usa as variáveis do ARCS Model of Motivational Design igualmente desenvolvidas por Keller. O IMMS foi projetado para medir a relação dos participantes com determinado dispositivo de média ou ferramenta introduzida durante a instrução e McGill (2012) adaptou-os especificamente para a robótica educativa. Este questionário contém dois tipos de questões: abertas e fechadas. Segundo Hill e Hill (2012), este tipo de questionário é útil quando se pretende obter informação qualitativa para complementar e contextualizar a informação quantitativa obtida pelas outras variáveis. As perguntas fechadas assumiram forma de afirmações em que os participantes tiveram de escolher um nível de concordância, segundo uma escala ordinal. Estas escalas admitem uma ordenação numérica das suas categorias, estabelecendo uma relação de ordem entre elas, contudo não é possível medir a magnitude das diferenças entre as categorias (Hill & Hill, 2012). Para Hill e Hill (2012) este tipo de questões para medir opiniões, atitudes e satisfações não devem obrigar o respondente a assumir uma resposta positiva ou negativa. Neste sentido, à semelhança de McGill (2012) para as questões de resposta fechada foi aplicada uma escala *Likert* de cinco níveis, onde 1 corresponde a não concordo totalmente, 2 a não concordo parcialmente, 3 a indiferente, 4 a concordo parcialmente e 5 a concordo totalmente. Em relação às questões abertas, estas foram analisadas segundo os critérios de comentário positivo, neutro ou negativo. Um comentário é considerado positivo quando expressa concordância com a questão, quando um comentário não expressa opinião ou é indiferente é considerado neutro, e por último um comentário é considerado negativo quando expressa crítica seja ela positiva ou negativa. Em termos de análise apenas serão estudados os comentários positivos e negativos, partindo do princípio que os comentários neutros não são significativos.

Assim como um marceneiro ao construir um móvel utiliza vários tipos de ferramentas, do mesmo modo um investigador produz informação na forma de conclusões aplicando diversas técnicas estatísticas aos dados de investigação (Hill & Hill, 2012). Quanto aos dados resultantes deste estudo, estes foram inseridos e analisados com recurso ao *software* de análise empresarial SPSS Statistics⁶ da IBM. A fiabilidade da classificação das respostas às questões abertas está ligada com o avaliador como fonte de inconsistência e dá-se o nome de fiabilidade de concordância. Para aumentar o grau de

⁶ Página do produto: <http://www-01.ibm.com/software/analytics/spss/products/statistics/downloads.html>

fiabilidade do estudo foi convidado outro avaliador para proceder igualmente à classificação e depois de várias revisões obteve-se um grau elevado de concordância entre dois avaliadores indicando um grau elevado de fiabilidade (Hill & Hill, 2012). As variáveis em estudo atitudes e perspetivas, e atenção, resultantes das questões fechadas, são do tipo variáveis latentes de investigação. Estas variáveis são usadas para representar uma variável que não pode ser observada nem medida diretamente, mas pode ser definida a partir de um conjunto de outras variáveis e são consideradas fiáveis se forem consistentes. Para medir a consistência das variáveis utilizaram-se diferentes métodos. Para a variável atitudes e perspetivas a consistência foi medida através do coeficiente de estabilidade temporal. Para isso, foi necessário aplicar a pergunta duas vezes à mesma amostra de pessoas e calcular a correlação entre as respostas. À semelhança com o estudo de McGill (2012), foi aplicado um pré e pós questionário. Este tipo de instrumento serve para medir o estado de espírito dos participantes, num momento inicial e posteriormente final, com o fim de analisar como esse estado afetou o seu desempenho nas tarefas atribuídas nos dois momentos distintos (Hill & Hill, 2012). Estatisticamente para calcular a consistência das questões emparelhadas da variável atitudes e perspetivas, calculou-se o *one sample t-test*, onde se o $p < 0.05$ a diferença temporal seria significativa. Para a mesma variável, procedeu-se a uma análise de Spearman testando a intensidade da relação entre as suas variáveis internas do pós questionário. Por outro lado, para análise da variável latente atenção não se optou por uma avaliação por coeficiente de estabilidade temporal, visto os participantes na fase inicial da intervenção não conseguirem ter opinião formada sobre o contributo do robô na no seu processo de aprendizagem. Optou-se antes pela estimativa de fiabilidade do tipo consistência interna através do cálculo do coeficiente alfa de Cronbach, onde se $\alpha > 0.80$ significa que as correlações entre as variáveis internas são altas, havendo evidências que as variáveis internas como um todo são representativas da variável latente, atenção.

6.2.4. Descrição do instrumento e processo de recolha de dados.

O processo de recolha de dados foi caracterizado por dois momentos distintos, um inicial e final, traduzidos num pré e pós questionário, tendo sido ambos disponibilizados aos participantes por administração direta em contexto de sala de aula (Quivy & Campenhoudt, 2008). Todas as questões do questionário foram baseadas no estudo de McGill (2012). O pré questionário (Anexo S) disponibilizado aos participantes no

primeiro dia de prática de ensino supervisionada, teve como objetivo medir as atitudes e percepções dos participantes face à programação com robôs. Segundo Hill e Hill (2012) os respondentes gostam de saber um pouco sobre o investigador que está a picar o questionário, como também de conhecer um pouco da natureza e objetivos de investigação. Os mesmos autores (2012) referem como elementos essenciais à introdução do questionário: um pedido de cooperação no preenchimento do questionário, a razão da aplicação do questionário, uma apresentação curta da natureza do questionário, o nome da instituição impulsionadora da investigação e uma declaração formal da confidencialidade das respostas. Seguindo estas orientações, foi elaborada a seguinte introdução para o pré questionário: “A professora Diana Oliveira no âmbito da Prática de Ensino Supervisionada integrante do Mestrado de Ensino de Informática da Universidade de Lisboa, está a desenvolver um estudo sobre o impacto da robótica educativa no ensino e aprendizagem de programação à disciplina de Aplicações Informáticas B, mais especificamente aos conteúdos: conceitos de programação e algoritmos. Este questionário constitui-se parte fundamental do estudo e destina-se a recolher informação sobre as atitudes e percepções dos alunos em relação a aprender programação com robôs. O questionário leva aproximadamente 5 minutos a responder. O preenchimento é individual e as respostas de cada inquirido serão salvaguardadas por confidencialidade, servindo apenas para fundamentação empírica do estudo. Desde já agradeço a colaboração. Nota: entenda-se que no questionário a palavra “ansioso” é usada não como sinónimo de “estar nervoso”, mas como “desejar veemente”. O questionário em si era composto por duas afirmações: “Estou ansioso para aprender a programar” e “Estou ansioso para aprender a programar com robôs”, às quais os respondentes atribuíram um grau de concordância.

Por outro lado, o pós questionário (Anexo W) aplicado no último dia de prática de ensino supervisionada, teve como objetivo medir as atitudes e percepções e atenção em relação à experiência de intervenção letiva, onde os respondentes tiveram contato com os robôs em contexto de aprendizagem. Seguindo as orientações de Hill e Hill (2012) enunciadas anteriormente sobre introdução de questionários, para o pós questionário foi elaborado a uma introdução semelhantes à do pré questionário, acrescentando que para o preenchimentos deste seriam facultados 15 minutos. Este questionário foi composto por duas dimensões, uma referente à variável atitudes e percepções e outra referente à variável atenção. A dimensão atitudes e percepções contempla duas afirmações: “Estou ansioso para aprender a programar”, “Estou ansioso para aprender a programar com robôs”.

Enquanto a dimensão atenção é composta por oito afirmações “Existe algo interessante que me chamou a atenção sobre aprender a programar com robôs”, “A forma como o robô foi usado para ensinar programação ajudou a manter minha atenção”, “Aprender a programar com o robô estimulou a minha curiosidade”, “O trabalho que realizamos para aprender a programar com o robô prendeu minha atenção”. Os respondentes face às afirmações, à semelhança do pré questionário atribuíram um grau de concordância. O questionário é composto igualmente por três questões abertas, que não são considerados como uma dimensão distinta a nível de variável, visto o seu objetivo é aprofundar a natureza dos dados recolhidos nas dimensões anteriores: atitudes e perceções, e atenção. As três questões de resposta aberta constituem-se: “Quais são os seus pensamentos e/ou sentimentos sobre aprender a programar?”, “Quais são os seus pensamentos e/ou sentimentos sobre o uso de robôs nesta disciplina?” e “Quais são os seus pensamentos e/ou sentimentos sobre como esta disciplina poderia ser melhorada?”.

6.3 Apresentação e análise de dados

No pré questionário obteve-se um total de 22 respostas o correspondente ao número de alunos da turma, enquanto no pós questionário obteve-se 19 respostas. Visto que para a análise coeficiente de estabilidade temporal, em variáveis latentes é importante manter a estabilidade da amostra, sendo que se pretende comparar o estado inicial com o final da variável, apenas foram consideradas para análise 19 respostas iniciais e finais, correspondentes aos mesmos participantes.

6.3.1 Dimensão atitudes e perceções.

Várias declarações relativas a atitudes e perceções foram colocadas aos participantes do estudo através de dois questionários, pré e pós, com o fim de determinar como se sentiram em relação à programação, aos robôs e a eles próprios. Os resultados apresentados no Quadro 15, revelam que inicialmente os participantes concordaram mais fortemente que estavam mais ansiosos para aprender a programar robôs com uma média de 4,26.

Quadro 15. Dimensão atitudes e percepções

	Pré-teste		Pós-teste		t-test
	Média	Desvio padrão	Média	Desvio padrão	p
Estou ansioso para aprender a programar.	4,21	0,71	4,05	0,97	0,49
Estou ansioso para aprender a programar com robôs.	4,26	1,05	3,95	0,91	0,15
t-test p	0,827		0,629		

Para testar se esta diferença é estatisticamente significativa, foi realizado um *one sample t-test* comparando a média da questão “Estou ansioso para programação com robôs”, com a média (constante) da questão “Estou ansioso para aprender a programar” ambos do pré-questionário. Para este ensaio de hipóteses foram analisadas as seguintes hipóteses:

- $H_0: \mu_1 - \mu_2 = 0$, a média das diferenças entre os valores da variável quantitativa nos dois momentos é nula, igual a zero;
- $H_a: \mu_1 - \mu_2 \neq 0$, a média das diferenças entre os valores da variável quantitativa nos dois momentos é nula, igual a zero;

Sendo μ_1 o nível médio de ansiedade para aprender a programar referente ao pré questionário e μ_2 o nível médio de ansiedade para aprender a programar com robôs igualmente do pré questionário. Os resultados do *one sample t-test* demonstram que o valor da prova do teste t foi igual a 0,222 e com valor p de 0,827 que é superior a 0,05, logo não se rejeita a hipótese nula, o que indica que não há diferenças estatisticamente significativas entre as duas medidas, ou seja, o nível de ansiedade para aprender a programar poderia ser igual ao nível de ansiedade para aprender a programar com robôs, estando os participantes igualmente ansiosos por aprender a programar e por aprender a programar com robôs. Salientando positivamente o facto de o grau de ansiedade, ou desejo veemente, ser em média 4 numa escala *Likert* de 5 níveis.

Analisando de novo o Quadro 15, pode observar-se que do pré para o pós questionário, tanto a média da afirmação “Estou ansioso para aprender a programar”, como a da “Estou ansioso para aprender a programar com robôs” diminuíram. Para testar estatisticamente a significância desta diminuição foram realizados dois *one-samples t-test* relacionando os valores iniciais e finais duas afirmação, segundo as seguintes hipóteses:

- $H_0: \mu_5 - \mu_6 = 0$, a média das diferenças entre os valores da variável quantitativa nos dois momentos é nula, igual a zero;
- $H_a: \mu_5 - \mu \neq 0$, a média das diferenças entre os valores da variável quantitativa nos dois momentos é nula, igual a zero;
- $H_0: \mu_7 - \mu_8 = 0$, a média das diferenças entre os valores da variável quantitativa nos dois momentos é nula, igual a zero;
- $H_a: \mu_7 - \mu_8 \neq 0$, a média das diferenças entre os valores da variável quantitativa nos dois momentos é nula, igual a zero;

Sendo μ_5 o nível médio de ansiedade para aprender a programar referente ao pré questionário e μ_6 o nível médio de ansiedade para aprender a programar referente ao pós questionário. O nível médio de ansiedade para aprender a programar com robôs referente ao pré questionário é representado por μ_7 e o nível médio de ansiedade para aprender a programar com robôs referente ao pós questionário por μ_8 . Em ambas as hipóteses o valor de significância é menor que 0,05, indicando que a descida não é significativa, logo não se rejeita a hipótese zero, o que leva a concluir que indicando que os alunos mantiveram o nível de ansiedade por programar e programar com robôs ao longo da intervenção. Em suma, segundo a análise os participantes revelaram o mesmo nível de ansiedade inicial face aprender a programar e aprender a programar com robôs e estes níveis não se alterarem ao longo da intervenção letiva.

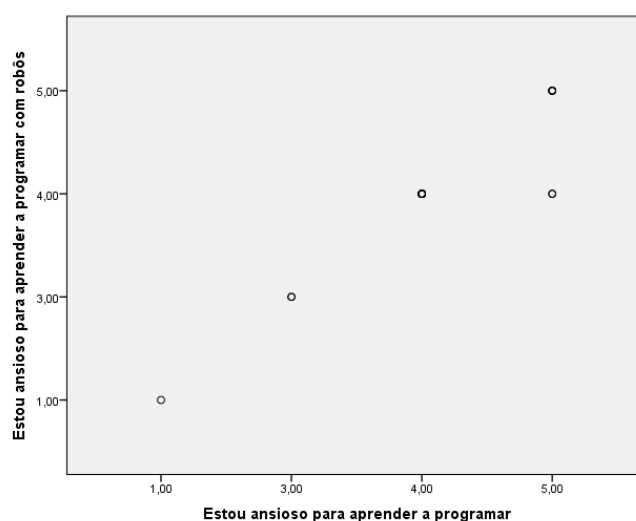


Figura 21. Correlação entre programar e programar com robôs

Foi realizada uma última análise sobre a correlação entre o nível de ansiedade para aprender a programar e aprender a programar com robôs, tendo em consideração os dados referentes ao pós questionário, partindo do princípio que nesta fase os participantes experienciaram a programação com robôs e assim terão uma opinião mais decisiva sobre a programação. Estatisticamente, visto a escala usada ser nominal, calculou-se o coeficiente de correlação de *Spearman* entre a questão “Estou ansioso por aprender a programar” e a questão “Estou ansioso por aprender a programar com robôs” (Figura 22). O valor encontrado para o coeficiente de correlação foi 0,876, podendo-se considerar que existe uma forte correlação entre as variáveis, visto o valor se encontrar próximo de 1, para além do que é um valor significativo, obtendo-se uma significância de $\rho=0.00$ que é inferior a $\rho<0.01$. Através da Figura 22 pode-se observar que à medida que aumenta o grau de ansiedade por programar, aumenta igualmente a ansiedade por programar com robôs.

6.3.2 Dimensão atenção.

Quadro 16. Dimensão atenção

	Pré-teste		Pós-teste	
	Média	Desvio padrão	Média	Desvio padrão
Existe algo interessante que me chamou a atenção sobre aprender a programar com robôs.	-	-	3,95	1,08
A forma como o robô foi usado para ensinar programação ajudou a manter minha atenção.	-	-	4,11	2,39
Aprender a programar com o robô estimulou a minha curiosidade.	-	-	4,00	0,94
O trabalho que realizámos para aprender a programar com o robô prendeu minha atenção.	-	-	3,79	0,92
A variedade de usos do robô ajudou a manter minha atenção.	-	-	3,79	0,85
Utilizar o robô para aprender a programar fez-me sentir recompensado por meu esforço.	-	-	3,74	1,05
Foi um prazer trabalhar com o robô para aprender a programar.	-	-	3,94	1,03
Eu gostei de usar o robô para aprender a programar tanto que eu gostaria de saber mais sobre ele.	-	-	3,68	0,95

Da observação dos resultados obtidos da aplicação do pós questionário referentes á dimensão atenção (Quadro 16), denota-se que todas as afirmações têm uma média positiva compreendida entre 3,68 e 4,11, indicando que os robôs tiveram uma influência positiva sobre os participantes despertando e mantendo a sua atenção. Para comprovar estatisticamente a significância deste estímulo, procedeu-se ao cálculo do coeficiente de alfa de Cronbach, com o resultado de $\alpha = 0,983$, sendo $\alpha > 0.80$ indica que existe um alto grau de confiabilidade da correlação entre as variáveis internas, logo existe evidência que as variáveis internas medem a mesma dimensão, neste caso a atenção. Assim, das respostas obtidas dos participantes em resultado das suas experiências o robô por si só, como objeto de aprendizagem prende a sua atenção e estimula a curiosidade. Salienta-se por outro lado que a forma como o robô foi usado e a variedade de usos do robô contribuíram para manter a atenção ao longo da intervenção, resultando numa aprendizagem prazerosa que fez com que alguns quisessem continuar a aprender programação com robôs nas aulas precedentes à intervenção.

6.3.3 Questões abertas.

O pós questionário era composto igualmente por três questões de resposta abertas. A questão "Quais são os seus pensamentos e / ou sentimentos sobre aprender a programar?" resultou em 13 comentários positivos, 3 neutros e 3 negativos (Anexo Y). Os comentários positivos incluíram “Aprecio bastante, programar é divertido, se for feito com um objetivo, eu espero que continuemos a programar como temos vindo a fazer”, “Tenho de admitir que achei bastante entusiástico, todo um processo de aprendizagem” e “Imensa curiosidade a possibilidade de aumentar a complexidade da programação e dos resultados que possam daí surgir”. Os comentários negativos incluíram " Prefiro a parte prática que a teórica”, tal como o desejo que ter aprendido uma linguagem de programação diferente.

A questão “Quais são os seus pensamentos e / ou sentimentos sobre o uso de robôs nesta disciplina?” resultou em 15 comentários positivos, 3 neutros e 1 negativa (Anexo Y). Os comentários positivos incluíram “Torna as coisas mais dinâmicas, se estivesse somente a copiar dados do quadro, ou escrever as explicações dos professores as aulas seriam aborrecidas. Trabalhar assim com os robôs é bastante melhor, e permite-nos ganhar experiência e conhecimentos teóricos, em vez de somente a teoria.”. Para além

disso, os participantes consideraram que o uso de robôs prendeu a sua atenção, e que aumentou o “interessante”, “gosto”, pela programação, tal como tornou o seu ensino mais criativo, “divertido”. À semelhança com a questão anterior, salienta-se como comentário negativo a expectativa de ter a aprendido uma linguagem de programação diferente.

A questão “Quais são os seus pensamentos e / ou sentimentos sobre como esta disciplina poderia ser melhorada?” resultou em 5 comentários positivos, 2 neutros e 12 negativos (Anexo Y). Os comentários positivos incluíram os participantes acharem que a disciplina foi bem lecionada e que não haveria nada a melhorar. Os comentários negativos são mais interessantes expressando sugestões de melhoramento, tais como, “mais tempo de aula ou mais espaço para realizar as experiências”, o desejo de “aprofundar mais os conteúdos”, não usar o robô basquetebolista e a preferência por aulas de componente prática.

6.4 Discussão e considerações

A análise de dados revelou um impacto positivo da robótica educativa tanto nas perceções e atitudes, como na atenção dos participantes, contrariando a revisão de literatura que afirmava que na generalidade os alunos não têm motivação intrínseca para estudar programação (Bereiter & Ng, 1991, citados de Gomes et al, 2008). Os participantes deste estudo apresentaram elevados níveis de expectativas em relação à programação, traduzido na variável atitudes e perceções. Um dos possíveis motivos deste acontecimento, pode ser a utilização da robótica educativa no ensino e aprendizagem de programação, visto os dados terem comprovado a existência de uma correlação entre as expectativas dos alunos face a programar e a programar com robôs.

No estudo de McGill (2012) os alunos estavam mais expetantes por programar com robôs do que simplesmente por programar, enquanto neste estudo os níveis de expectativa entre programar e programar com robôs eram os mesmos, salientando-se que estes níveis mantiveram-se ao longo da intervenção letiva. Este facto pode dever-se aos diferentes interesses das amostras, uma vez que o estudo de McGill (2012) foi aplicado no ensino superior e este foi realizado no ensino secundário. Sobre a amostra deste estudo através do capítulo dois Contexto da Prática de Ensino Supervisionada, mais especificamente no tópico Caracterização da turma sabia-se que os alunos atribuíam alguma importância à unidade de ensino-aprendizagem Introdução à Programação, uma

média correspondente a 3 numa escala *Likert* de 4. Este valor pode estar relacionado com o facto de alguns alunos terem intenção de seguir estudo na área de informática e por isso considerarem a programação no geral como algo importante e estarem predispostos a aprender a programar.

Das diversas variáveis estudadas por McGill (2012) a atenção foi a única que se traduziu significativa no processo de ensino e aprendizagem de programação segundo a análise dos dados correspondentes às perspetivas dos alunos. Neste estudo a variável atenção também se revelou significativa no ensino a aprendizagem de programação, e aos resultados estatísticos acrescentam-se algumas opiniões dos alunos tais como: “Os robôs ajudam a prender a atenção”, “Torna as coisas mais dinâmicas”, “Os robôs foram, de certa forma, a forma mais divertida que me ocorre de testar o aprendido na bela linguagem que é a programação” e “Acho interessante visto que segundo o programa iríamos aprender uma básica iniciação à programação. Nada mais empolgante que robôs!”.

7. Reflexão da Prática de Ensino Supervisionada

Ser professora sempre foi um sonho para mim e assim que surgiu oportunidade abracei a atividade docente do grupo 550. O primeiro ano que lecionei não foi fácil, visto ter muitas turmas de cursos e disciplinas variadas, desde 5º, 6º, 7º, 9º CEF Cozinha e EFA. Com o passar do tempo senti a necessidade de me inteirar melhor sobre estes cursos e fiz uma formação em Mediadores e Formadores EFA. Cedo me apercebi que esta formação constituiu uma pequena ajuda e que havia ainda muito que aprender e aprimorar. Nesse ano letivo, apesar de ter dado o meu melhor e considerar que realizei um bom trabalho, senti que não consegui diferenciar as minhas aulas, especialmente quando tinha turmas da mesma disciplina e ano. No segundo ano que lecionei fiquei dedicada a uma turma de 9º ano CEF Informática. Lecionava disciplinas tais como Gestão de Bases de Dados, Aplicações de Escritório, Instalação e Configuração de Redes e Internet entre outras. De facto, é um grande desafio, ter a capacidade de lecionar disciplinas tão variadas dentro da informática. A verdade é que apesar de ser a minha área sentia que o meu conhecimento era muito maduro para ser transmitido aos alunos assim em bruto, e os livros que lia da área não era se adaptavam às suas faixas etárias, nem eram muito estimulantes a nível de abordagens. Nesta fase da minha vida profissional senti necessidade de obter uma formação mais profunda e específica sobre como lecionar e decidi inscrever-me no Mestrado de Ensino de Informática da Universidade de Lisboa, na esperança de aprender e trocar experiências com quem lidava diretamente com estes cursos, turmas e ensino.

Desde o início da frequência do mestrado desenvolvi um interesse pelo ensino de programação. O meu trabalho final à disciplina de Didática I foi uma análise curricular da disciplina de Programação e Sistemas de Informação [PSI] e seu enquadramento de com a Aprendizagem Baseada por Problemas [ABP]. Em IPP II analisei estatísticas sobre o ensino de programação e quais as maiores dificuldades no seu ensino. Em Didática II realizei um trabalho sobre o ensino de programação e a robótica educativa, tendo sido este trabalho o impulsionador da publicação dos artigos: Uma proposta de ensino-aprendizagem de programação utilizando robótica educativa e storytelling (Oliveira et al., 2012) e Uma proposta de ensino-aprendizagem da disciplina Aplicações Informáticas B utilizando robótica educativa e storytelling (Ferreira et al., 2013) dos quais fiz parte. Toda esta aprendizagem e experiência contribuiu para o planeamento e concretização desta prática de ensino supervisionada. O autor de maior referência foi Papert (1993) que

defende que o papel da escola não deve ser treinar pessoas para desempenhar uma determinada função para o resto da vida, mas estimular nos alunos a capacidade de aprender novas habilidades, assimilar novos conceitos, avaliar novas situações e lidar com o inesperado. Apesar de não existem alunos, nem turmas iguais, o que torna impossível conceber soluções únicas, Rusk e colaboradores (2008) defendem que metodologias de aprendizagem baseadas por resolução de problemas, onde o problema base é deixado em aberto, proporcionam situações de aprendizagem onde os alunos exploram, elaboram, refazem, refletem soluções ultrapassando obstáculos e lidando com o inesperado, ao mesmo tempo que personalizam o problema às suas necessidades de aprendizagem. Papert (1993) aponta como o grande problema do ensino a falta de motivação dos estudantes que, em parte, é determinada pelo facto de os jovens não perceberem a utilidade do que estão a aprender. Seguindo as pisadas dos autores referidos (Papert, 1993; Rusk et al., 2008) a intervenção letiva foi planeada em torno da robótica educativa como ferramenta e objeto de aprendizagem de programação. A nível pessoal como gosto de *design* e de banda desenha, até mesmo participei no evento Games 2006 na categoria de Melhor Narrativa/Argumento de Jogo, logo me simpatizei com os estudos de Rusk e colaboradores (2008) que associam o *storytelling* à robótica educativa. Achei interessante propor aos alunos desafios em forma de banda desenhada, onde neste caso a Mulher Invisível ajuda-os a aprender a controlar os robôs, na expectativa de possam salvar a cidade. Desta forma, não existe um ambiente em que a professora manda e exige, antes os alunos são envolvidos em desafios que promovem a aprendizagem.

Este tipo de metodologia de aprendizagem é desafiante e exige um grande nível de confiança. Visto a estratégia optada durante o desenrolar dos desafios ser fazer rondas sucessivas às equipas, onde podia assumir quatro papéis: apenas apreciar como comunicavam, ajudar as equipas a refletir sobre o seu trabalho, tirar dúvidas específicas, ou questionar as equipas sobre aspetos específicos. O que acontece é que o professor passa muito tempo de costas para a turma em geral, o que faz com que se o desafio não for estimulante e interessante para os alunos, quando não estivermos por perto eles não vão trabalhar. Nesse aspeto, os robôs cativaram a atenção dos alunos, tanto que na primeira aula as equipas cedo descobriram como se programava o robô, e nem sabia o que o desafio pedia para fazer, o que denota algum fascínio pelo robô em si, que vai de encontro com os estudos de Kumar e Meeden (1998). Recordando a intervenção letiva, foi muito interessante notar que os alunos tanto na programação com i-Brick como em NXT-G, sem recurso a manuais, nem instruções específicas de como se programa,

aprendiam através da experimentação o que era a programação. Isto apenas foi possível devido ao uso de linguagens de programação visuais, onde os problemas de sintaxe foram atenuados, dando tempo e espaço para os alunos se concentrarem na resolução de problemas através do pensamento computacional e de uma forma livre aprenderem para além de desenvolverem programas. Os alunos revelavam vontade em superar os seus desafios. A análise de dados revelou que os alunos tinham grandes expectativas em relação a aprender a programar com robôs e que não ficaram desiludidos, mantendo o seu nível de expectativa até ao final da intervenção. Revelando que gostaram do desafio 1, desafio 2 e da maneira como foi abordado.

Estou expetante por voltar a lecionar, sinto-me mais completa a nível profissional, sinto que consigo ser um elemento mais ativo e participativo da comunidade docente. Esta experiência ensinou-me a identificar problemas de ensino e aprendizagem, a pesquisar metodologias e ferramentas de minimizem esses dilemas, fazer planificações de implementação, a ter coragem de concretiza-las, a refletir e aprender com elas. Ensinou-me a não desistir e a acreditar nas palavras de Papert (1980) de que com ferramentas e estratégias de ensino adequadas é possível em certa medida promover alunos peritos em programação.

Referências

- Abreu, C., & Loureiro, C. (Dezembro, 2007) Aprendizagem por Resolução de Problemas – Uma experiência pluridisciplinar e multicultural. Referencia, 2 (5), páginas.
- Aguilar, L. (2008). Fundamentos da Programação – Algoritmos, estruturas de dados e objetos (3rd ed.) (P.H.C. Valle, Trad.). São Paulo: Mc-Graw Hill.
- Ala-Mutka, K. (2003). Problems in Learning and Teaching Programming - A literature Study for Developing Visualizations in the Codewitz-Minerva Project. Codewitz Needs Analysis. Retirado de <http://www.codewitz.net/minerva/>.
- Alarcão, I. (2000). *Escola Reflexiva e Supervisão. Uma Escola em Desenvolvimento e Aprendizagem*. Porto: Porto Editora.
- Alimisis, D. (2009). *Teacher Education on Robotics-Enhanced Constructivist Pedagogical. Methods*. Athens: School of Pedagogical and Technological Education (ASPETE)
- Arends, R. (2008). *Aprender a ensinar* (A. Faria, Trad.). 7ª ed. Madri: McGraw-Hill (Obra original publicada em 2007).
- Becker, F. (2009). O que é construtivismo?. Desenvolvimento e Aprendizagem sob o Enfoque da Psicologia II. UFRGS – PEAD 2009/1.
- Benitti, F. (2012). Exploring the educational potential of robôics in schools: A systematic review. Computers e Education, 58, 978-988.
- Bernardo, M. (2012). A robótica educativa aplicada na consolidação de conhecimentos na disciplina de Linguagens de Programação. Tese de mestrado apresentada à Universidade de Lisboa, Lisboa.
- Boshernitsan, M., & Downes, M. (Dezembro de 2004). Visual Programming Languages: A Survey. Report No. UCB/CSD-04-1368.
- Carvalho, A., Salles, F., Guimarães, M. (2002). *Desenvolvimento e Aprendizagem*. Belo Horizonte: EditoraUFMG.
- Carvalho, J., Lima, R., Fernandes, S. (2008). Aprendizagem em engenharia: projetos e equipas interdisciplinares. Retirado em 23 de Março de http://pessoais.dps.uminho.pt/jdac/ensino/Artigo_PIEI.pdf.
- Cervo, A. & Bervian, P. (1996). *Metodologia científica*. 4.ed. São Paulo: Makron Books.
- Cook, J. & Cook, L. (1999). How technology enhances the quality of student learning. *Quality Progress*.
- Cormen, Leiserson, Rivest & Stein, (2009). *Introduction to Algorithms*. MIT Press.
- Cortazzi, M. (1993). Narrative Analysis. London: The Falmer Press.
- Costa, J. (2012). A robótica educativa na aprendizagem da programação. Tese de mestrado apresentada à Universidade de Lisboa, Lisboa.
- CSTA (2012, Novembro). The National Imperative for K-12 Computer Science Education. Artigo apresentado do Microsoft Race to the Future Panel. Retirado de http://csta.acm.org/Advocacy_Outreach/sub/PresentationFiles/MicrosoftPanel_Step_henson.pdf.
- Decreto-Lei n.º 272/2007. Diário da República – 1ª Série – N.º 143 – 26 de Julho. Ministério da Educação. Lisboa.
- Decreto-Lei n.º 50/2011a. Diário da República – 1ª Série – N.º 70 – 8 de Abril. Ministério da Educação e Ciência. Lisboa.
- Decreto-Regulamentar n.º 244/2011b. Diário da República, 1ª Série — N.º 118 — 21 de Junho de 2011.
- Decreto-Regulamentar n.º 26/2012. Diário da República – 1ª Série – N.º 37 – 21 de Fevereiro. Ministério da Educação e Ciência. Lisboa.

- Escola Secundária de Camões [ESC] (2010). Projeto Educativo 2010/2013. Retirado de http://escamoes-web.sharepoint.com/Documents/pe_2010_2013.pdf.
- Escola Secundário de Camões [ESC] (2012). Proposta de oferta da disciplina de Aplicações Informáticas B para o ano letivo 2012/2013.
- Escola Secundária de Camões [ESC] (2013). Relatório da Direção 1º Trimestre do Ano Letivo 2012/2013. Retirado de <http://portal.escamoes.pt/images/docs/rel2012final.pdf>.
- Ferreira, S., Oliveira, D., Celestino, H., & Abrantes, P. (2013, Maio). Uma proposta de ensino-aprendizagem da disciplina Aplicações Informáticas B utilizando robótica educativa e storytelling. Artigo apresentado na Conferência Internacional Investigação, Práticas e Contextos em Educação, Leiria, Portugal.
- Figueiredo, A., & Afonso, A. (2006). Context and Learning: A Philosophical Framework. In A. D. Figueiredo and A. P. Afonso (Eds.) *Managing Learning in Virtual Settings: the Role of Context*, Hershey, PA, USA: Information Science Publishing, pp. 1-22.
- Fuller, U., Johnson, C., Ahoniemi, T., Cukierman, D., Hernán-Losada, I., Jackova, J., Lahtinen, E., Lewis, T., Thompson, D., Riedesel, C., & Thompson, E. (2007). Developing a computer science-specific learning taxonomy. *ACM SIGCSE Bulletin*, 39(4), 152-170. doi: 10.1145/1345375.1345438.
- Galvão, C. (2005). Narrativas em educação. *Ciência & Educação*, v. 11, n. 2, p. 327-345.
- Gaspar, L. (2007). Os robôs nas aulas de informática. Tese de mestrado apresentada à Universidade da Madeira, Funchal.
- Gomes, A., Henriques, J., & Mendes, A. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação e Tecnologia*, 1, 93-103.
- Gomes, G. (2012). A robótica educativa no ensino da programação. Tese de mestrado apresentada à Universidade de Lisboa, Lisboa.
- Gomes, G., Martinho, J., Bernardo, M., Matos, F., & Abrantes, P. (2012, Dezembro). Dificuldades na aprendizagem da programação no ensino profissional – Perspetiva dos alunos. Artigo apresentado na II Conferência Internacional TIC e Educação, Lisboa, Portugal.
- Griffin, T. (2010). *The Art of LEGO® MINDSTORMS® NXT-G Programming*. San Francisco: No Starch Press.
- Hill, M. & Hill, A. (2012). *Investigação por Questionário*. Lisboa: Edições Sílabo.
- Imberman, S. (2004). An Intelligent Agent Approach for Teaching Neural Networks Using LEGO Handy Board Robôs, *JERIC, ACM Journal of Educational Resources in Computing* 4(3):1-12.
- Johnson, J. (2003). Children, robôics and education. In *Proceedings of 7th international symposium on artificial life and robôics* (Vol. 7, pp. 16–21), Japan: Oita.-> confirma esta referência, está um misto de artigo de Conferência e capítulo de livro
- Kim S., & Jeon J. (2007, Outubro). Programming LEGO Mindstorms NXT with visual programming. Artigo apresentado na International Conference on Control, Automation and Systems 2007, Seoul, Coreia do Sul.
- Koski, M., Kurhila, J., & Pasanen, T. A. (2008, Novembro). Why Using Robôs to Teach Computer Science can be Successful Theoretical Reflection to Andragogy and Minimalism. Paper presented at Koli Calling '08, Koli.
- Klassner Frank (2002). A case study of LEGO Mindstorms™ suitability for Artificial Intelligence and robôics courses at the college level, *SIGCSE, Special Interest Group on Computer Science Education, Proc. of the 33rd SIGCSE Technical Symposium on Computer Science Education*, Covington, Kentucky, USA, 8-12.

- Kumar, A. (2001). Using Robôs in an Undergraduate Artificial Intelligence Course: An Experience report, 31st Annual ASEE/IEEE Frontiers in Education Conference, Reno, Nevada, USA, 2:10-14.
- Kumar, D. & Meeden, L. (1998). A Robô Laboratory for Teaching Artificial Intelligence, SIGCSE, Special Interest Group on Computer Science Education, Proc. of the 29th SIGCSE technical symposium on Computer Science Education, Atlanta, Georgia, USA, 341-344. Retirado de http://mainline.brynmawr.edu/Robôs/ResourceKit/Paper.html#Acknowledgement_
- Kurland, D.M., Pea, R.D., Clement, C., & Mawby, R. (1989). A study of the development of programming ability and thinking skills in high school students. In: Soloway, E., Spohrer, J.C. (Eds.), *Studying the Novice. Programmer*. London, Lawrence Erlbaum Associates, 83–112.
- Lego (2009). Mindstorms Education NXT User Guide. Retido de <http://mindstorms.lego.com/en-us/support/buildinginstructions/8547/8547%20User%20Guide%20English.aspx>
- Mack, N., Woodsong, C., Macqueen, K., Guest, G. & Namey, E. (2005). *Qualitative Research Methods: A Data Collector's Field Guide*. North Carolina: Family Health International.
- Maliuk, K. (2009). Robótica educacional como cenário investigativo nas aulas de matemática. Local: Editora.
- Martins, J., & Cravo, M. (2011). *Fundamentos da programação utilizando múltiplos paradigmas*. Lourinhã: IST Press.
- Nóvoa, A. (2001). O Professor Pesquisador e Reflexivo. Retirado de http://www.tvebrasil.com.br/salto/entrevistas/antonio_novoa.htm.
- Oliveira, D., Ferreira, S., Celestino, H., Ferreira, S., & Abrantes, P. (2012, Dezembro). Uma proposta de ensino-aprendizagem de programação utilizando robótica educativa e storytelling. Artigo apresentado no II Congresso Internacional TIC e Educação, Lisboa, Portugal.
- Papert, S. (1980). *Mindstorms. Children, Computers and Powerful Ideas*. Basic Books, Inc. Publishers, New York.
- Papert, S. (1993). *The children's machine: rethinking school in the age of the computer*. Nova Iorque: Harvester/Wheatsheaf.
- Phillips, P. (2009). Computational thinking a problem-solving tool for every classroom. Retirado de <http://csta.acm.org/Resources/sub/ResourceFiles/CompThinking.pdf>.
- Pinto, M., Dias, P., & João, S (2006). Programa de Aplicações Informáticas B de 12º ano. Ministério da Educação.
- Pinto, M., Dias, P., & João, S (2009). Programa de Aplicações Informáticas B de 12º ano. Ministério da Educação
- Quivy, R., & Campenhoudt, L. V. (2008). *Manual de Investigação em Ciências Sociais*. 5ª Ed. Lisboa: Gradiva.
- Reis, P. (2011). Observação de aulas e avaliação do desempenho docente. Lisboa: Ministério da Educação – Conselho Científico para a Avaliação de Professores.
- Ribeiro, C. (2006). RobôCarochinha: Um Estudo Qualitativo sobre a Robótica Educativa no 1º ciclo do Ensino Básico. Tese de mestrado apresentada à Universidade do Minho, Braga.
- Ribeiro, A., & Ribeiro, L. (1990). *Planificação e avaliação do ensino-aprendizagem*. Lisboa: Universidade Aberta.
- Roldão, M. (2009). *Estratégias de Ensino*. Vila Nova de Gaia: Fundação Manuel Leão.
- Rusk, N., Resnick, M., Berg, R., & Pezalla-Granlund, M. (2008). New Pathways into Robotics: Strategies for Broadening Participation, *Journal of Science Education and Technology*, 17(1), 59-69. doi: 10.1007/s10956-007-9082-2.

- Saeli, M., Perrenet, J., Jochems, W., & Zwaneveld B. (2011). Teaching Programming in Secondary School: A Pedagogical Content Knowledge Perspective. *Informatics in Education*, Vol. 10, No. 1, 73–88.
- Santos, L. (2003). Avaliar competências: uma tarefa impossível? *Educação e Matemática*, 74, 16-21.
- Santos, L. & Pinto, J. (2003). O que os alunos pensam sobre a avaliação? *Educação e Matemática*, 74.
- Sebesta, R. (2003). *Concepts of programming languages*. 5ª Ed., Addison-Wesley.
- Serafini, G. (2011). Teaching programming at primary schools: visions, experiences, and long-term research prospects. Proceedings of the 5th international conference on Informatics in Schools: situation, Evolution and Perspectives Springer-Verlag, Berlin, Heidelberg, pp. 143.
- Tuckman, B. (2005). *Manual de Investigação em Educação*. 3ª Ed., Lisboa: Fundação Calouste Gulbenkian.
- Usategui, J., & Leon, J. (1986). *Guia Fácil de Robótica*. Madrir: Paraninfo.
- Varanda, F. (2012). Robótica educativa no ensino de subprogramas. Tese de mestrado apresentada à Universidade de Lisboa, Lisboa.
- Wing, J. (2006, Março). Computational Thinking. *Communications of the ACM*, Vol. 49, No. 3.
- Zilli, S. (2004). A Robótica Educacional no Ensino Fundamental: Perspetivas e Prática. Dissertação de Mestrado apresentada à Universidade Federal de Santa Catarina, Florianópolis.

Anexos

Anexo A: Questionário de caracterização da turma

Questionário de Caraterização da Turma



Este questionário faz parte de um estudo preliminar para a preparação de uma intervenção pedagógica a decorrer na turma.

O objetivo deste questionário é conhecer os alunos para a intervenção pedagógica ir de encontro às necessidades, apetências e expetativas dos alunos.

Obrigado pela colaboração.

***Obrigatório**

Nome *

Endereço de Correio Eletrónico *

1. Quando terminar o 12.º ano pretende continuar os estudos? *

- ☐ Sim
☐ Não

1.1 Se respondeu sim na questão 1, em que área pretende fazê-lo?

2. Quais as disciplinas que mais gosta? *

- ☐ Português
☐ Educação Física
☐ Matemática A
☐ Biologia
☐ Física
☐ Química
☐ Psicologia B
☐ Aplicações Informáticas B

3. Quais as disciplinas que menos gosta? *

- ☐ Português
- ☐ Educação Física
- ☐ Matemática A
- ☐ Biologia
- ☐ Física
- ☐ Química
- ☐ Psicologia B
- ☐ Aplicações Informáticas B

3.1 Porquê? *

4. Quantas vezes no seu percurso escolar já reprovou? *

- ☐ 0
- ☐ 1
- ☐ 2
- ☐ 3 ou mais vezes

4.1 Se já reprovou, em que ano(s) de escolaridade reprovou e porquê?

5. Porque escolheu a disciplina de opção Aplicações Informáticas B? *

- ☐ Gosto pela Informática
- ☐ Influência dos colegas
- ☐ Influência dos pais
- ☐ Influência dos professores

☐ Outra:

6. Qual o grau de importância que atribui à disciplina de Aplicações Informáticas B? *

1 2 3 4

Nada Importante ☐ ☐ ☐ ☐ Muito Importante

7. Qual o grau de importância que atribui à unidade de Introdução à Programação? *

1 2 3 4

Nada Importante ☐ ☐ ☐ ☐ Muito Importante

8. Qual o seu grau de interesse em relação à utilização de robots na unidade de Introdução à Programação? *

1 2 3 4

Nada Importante ☐ ☐ ☐ ☐ Muito Importante

9. Já alguma vez programou? *

- ☐ Sim.
☐ Não. Avance para a questão 10.

9.1. Em que contexto programou?

- ☐ Na escola
☐ Em casa
☐ Outra:

9.2. Que linguagens de programação já utilizou?

- ☐ C
☐ Java
☐ NXT-G
☐ C++
☐ C#
☐ .NET

☐ Pascal

☐ Outra:

10. Já alguma vez utilizou um robot? *

☐ Sim.

☐ Não. Avance para a questão 11.

10.1 Qual o robot que utilizou?

☐ Lego Mindstorms

☐ Arduino

☐ Tartaruga

☐ Outra:

11. Quando tem dúvidas, a quem recorre? *

☐ Aos colegas.

☐ Aos professores.

☐ Aos familiares.

☐ À Internet.

☐ Aos livros.

☐ Outra:

Anexo B: Autorizações para a realização da prática de ensino supervisionada

Exmo. Senhor Diretor da
Escola Secundária de Camões

Diana Oliveira, Honorina Celestino e Susana Ferreira, alunas do 2º ano Mestrado em Ensino da
Informática da Universidade de Lisboa, orientadas

vêm requerer a V. Ex.ª autorização para procederem à recolha de dados
dos alunos do 12º ano do Curso de Ciências e Tecnologias, turmas A, B e E (alunos de Aplicações
Informáticas B), nomeadamente respostas a questionários ou entrevistas, e filmagens e/ou gravação. Os
referidos registos visam a obtenção de dados num estudo relacionado com a caracterização dos
alunos. Serão salvaguardados todas as questões éticas e legais de recolha de dados.

O Estudo surge no âmbito da intervenção pedagógica que se realizará ao longo do presente ano
letivo, nas referidas turmas. O trabalho de intervenção terminará com a elaboração de um relatório
final das disciplinas de Introdução à Prática Profissional III e IV, do Mestrado em Ensino de
Informática, do Instituto de Educação da Universidade de Lisboa.

Os dados recolhidos terão um carácter confidencial, servindo apenas para a fundamentação da
parte empírica do trabalho.

Oportunamente, serão informados os respetivos Diretores de Turma e serão solicitadas aos
Encarregados de Educação as devidas autorizações para a participação dos seus educandos neste
trabalho.

Lisboa, 05 de Novembro de 2012

Pede deferimento



As Professoras


(Diana Oliveira)


(Honorina Celestino)


(Susana Ferreira)

Exmo(a). Senhor(a) Encarregado(a) de Educação

Diana Oliveira, Honorina Celestino e Susana Ferreira, alunas do 2º ano Mestrado em Ensino da Informática da Universidade de Lisboa, orientadas

, vêm requerer a V. Ex.ª autorização para procederem à recolha de dados, do seu educando, nomeadamente na resposta a questionários ou entrevistas e filmagens e/ou gravação. Os referidos registos visam a obtenção de dados num estudo relacionado com a caracterização dos alunos. Serão salvaguardados todas as questões éticas e legais de recolha de dados.

O Estudo surge no âmbito da intervenção pedagógica que se realizará ao longo do presente ano letivo. O trabalho de intervenção terminará com elaboração de um relatório final das disciplinas de Introdução à Prática Profissional III e IV, do Mestrado em Ensino da Informática, do Instituto de Educação da Universidade de Lisboa. Vimos solicitar autorização a V. Ex.ª para que nos faculte a participação e o contributo do seu educando neste estudo, de acordo com o que foi referido.



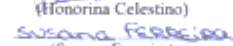
Os dados recolhidos terão um carácter **confidencial**, servindo apenas para a fundamentação da parte empírica do trabalho.

Informamos, ainda, que já pedimos autorização à Direção da Escola e à Direção de Turma.

Agradecemos desde já a atenção dispensada.

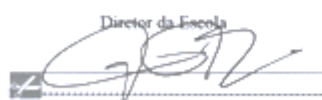
Com os melhores cumprimentos,

As Professoras


(Diana Oliveira)

(Honorina Celestino)

(Susana Ferreira)

Lisboa, 05 de Novembro de 2012

Tomei conhecimento:


Diretor da Escola


Diretor(a) de Turma

Eu, _____ Encarregado(a) de Educação do(a) aluno(a) _____, n.º _____, da turma _____, do _____, autorizo o meu educando a contribuir com a sua participação para o trabalho das alunas Diana Oliveira, Honorina Celestino e Susana Ferreira.

Lisboa, 02 de Novembro de 2012

Assinatura do Encarregado de Educação

Exmo(a). Senhor(a)

Exmo(a). Senhor(a) Encarregado(a) de Educação

Diana Oliveira, Honorina Celestino e Susana Ferreira, alunas do 2º ano Mestrado em Ensino da Informática da Universidade de Lisboa, orientadas

, vêm requerer a V. Ex.ª autorização para procederem à recolha de dados, do seu educando, nomeadamente na resposta a questionários ou entrevistas e filmagens e/ou gravação. Os referidos registos visam a obtenção de dados num estudo relacionado com a caracterização dos alunos. Serão salvaguardados todas as questões éticas e legais de recolha de dados.

O Estudo surge no âmbito da intervenção pedagógica que se realizará ao longo do presente ano letivo. O trabalho de intervenção terminará com elaboração de um relatório final das disciplinas de Introdução à Prática Profissional III e IV, do Mestrado em Ensino da Informática, do Instituto de Educação da Universidade de Lisboa. Vimos solicitar autorização a V. Ex.ª para que nos faculte a participação e o contributo do seu educando neste estudo, de acordo com o que foi referido.

Os dados recolhidos terão um carácter **confidencial**, servindo apenas para a fundamentação da parte empírica do trabalho.

Informamos, ainda, que já pedimos autorização à Direção da Escola e à Direção de Turma.

Agradecemos desde já a atenção dispensada.

Com os melhores cumprimentos,

As Professoras


(Diana Oliveira)


(Honorina Celestino)


(Susana Ferreira)

Lisboa, 05 de Novembro de 2012

Tomei conhecimento:


Diretor da Escola


Diretor(a) de Turma

Eu, _____ Encarregado(a) de Educação do(a)
aluno(a) _____, n.º _____, da turma _____, do _____, autorizo o
meu educando a contribuir com a sua participação para o trabalho das alunas Diana Oliveira, Honorina
Celestino e Susana Ferreira.

Lisboa, 02 de Novembro de 2012

Assinatura do Encarregado de Educação

Exmo(a). Senhor(a)

Exmo(a). Senhor(a) Encarregado(a) de Educação

Diana Oliveira, Honorina Celestino e Susana Ferreira, alunas do 2º ano Mestrado em Ensino da Informática da Universidade de Lisboa, orientadas

vêm requerer a V. Ex.ª autorização para procederem à recolha de dados, do seu educando, nomeadamente na resposta a questionários ou entrevistas e filmagens e/ou gravação. Os referidos registos visam a obtenção de dados num estudo relacionado com a caracterização dos alunos. Serão salvaguardados todas as questões éticas e legais de recolha de dados.

O Estudo surge no âmbito da intervenção pedagógica que se realizará ao longo do presente ano letivo. O trabalho de intervenção terminará com elaboração de um relatório final das disciplinas de Introdução à Prática Profissional III e IV, do Mestrado em Ensino da Informática, do Instituto de Educação da Universidade de Lisboa. Vimos solicitar autorização a V. Ex.ª para que nos faculte a participação e o contributo do seu educando neste estudo, de acordo com o que foi referido.

Os dados recolhidos terão um carácter **confidencial**, servindo apenas para a fundamentação da parte empírica do trabalho.

Informamos, ainda, que já pedimos autorização à Direção da Escola e à Direção de Turma.

Agradecemos desde já a atenção dispensada.

Com os melhores cumprimentos,

As Professoras

Diana Oliveira
(Diana Oliveira)

Honorina Celestino
(Honorina Celestino)

Susana Ferreira
(Susana Ferreira)

Lisboa, 05 de Novembro de 2012

Tomei conhecimento:

Diretor da Escola

[Assinatura]

Diretor(a) de Turma

Sede F. B.

Eu, _____ Encarregado(a) de Educação do(a) aluno(a) _____, n.º _____, da turma _____, do _____, autorizo o meu educando a contribuir com a sua participação para o trabalho das alunas Diana Oliveira, Honorina Celestino e Susana Ferreira.

Lisboa, 02 de Novembro de 2012

Assinatura do Encarregado de Educação

Exmo(a). Senhor(a)

Anexo C: Formulário de Observação de Aulas do Professor Cooperante



Formulário de Observação de Aulas do Professor Cooperante



Organização da sala de aula

1. A sala permite uma disposição de mesas e cadeiras flexível? ☒ Sim ☐ Não
2. A que distância os alunos se sentam uns dos outros? ☒ Perto ☐ Longe
3. A que distância se encontra o professor em relação aos alunos? ☒ Perto ☐ Longe
4. A sala tem espaço de trabalho suficiente? ☒ Sim ☐ Não
5. Os alunos escolhem os lugares onde se sentam em cada aula? ☐ Sim ☒ Não
6. Que recursos estão disponíveis na sala de aula? Computadores, Projetor, Quadro branco e Robots.

Gestão da sala de aula

1. Os alunos saem dos seus lugares no decorrer da aula? ☐ Sim ☒ Não
2. Os alunos estão familiarizados com as regras de funcionamento da sala de aula?
☒ Sim ☐ Não
3. Como é que os alunos estão organizados para trabalhar?
☐ Individualmente ☒ Em grupo de 2 elementos
4. Os alunos estão adaptados à organização de trabalho utilizada? ☒ Sim ☐ Não

Discurso do professor





1. Que tipo de interação professor-aluno existe na sala de aula?
☐ Individual ☒ Grupo ☐ Plenário
2. O professor dá tempo aos alunos para pensarem depois de fazer uma pergunta?
☒ Sim ☐ Não

Discurso dos alunos

1. Com que frequência os alunos fazem perguntas? ☐ Baixa ☒ Média ☐ Elevada
2. Há conversas entre os alunos? ☐ Apenas sobre a aula ☒ Outros assuntos
3. Todos os alunos recebem a mesma atenção do professor? ☒ Sim ☐ Não

Anexo D: Kit da Lego Mindstorms NXT

Aa informações contidas neste anexo foram retiradas do *site* oficial da Lego® Mindstorms® NXT, tal como do manual de instruções do *kit* da Lego® Mindstorms® NXT 2.0.

9841 NXT Intelligent Brick	
	<p>32 bits ARM7 microprocessador</p> <p>Suporte para comunicação wireless por Bluetooth</p> <p>1 porta USB 2.0</p> <p>3 portas de input</p> <p>3 portas de output</p> <p>Alimentado por 6 pilhas do tipo AA de 1.5 volts, ou bateria recarregável de lítio, modelo #9798 da LEGO®.</p>
9843 Sensor de toque	
	<p>O sensor de toque reage ao pressionar e liberar do botão. O kit contém dois sensores de toque que podem ser usados em simultâneo, sendo que o I-Brick permite detetar se apenas foi pressionado um botão ou dois ao mesmo tempo.</p>
9844 Sensor de luz e cor	
	<p>O sensor de luz permite que o robot distinga entre claro e escuro, bem como determinar a intensidade da luz em um quarto ou a intensidade de luz de cores diferentes.</p>
9845 Sensor de som	
	<p>O sensor de som é capaz de medir os níveis de ruído em ambos dB (decibéis) e dBA (cerca de 3-6 kHz frequências em que o ouvido humano é mais sensível), bem como o reconhecimento de padrões de som e identificar as diferenças de tom.</p>

9846 Sensor de ultrassom	
	<p>O sensor de ultrassom é capaz de detectar um objeto e medir a sua proximidade em polegadas ou centímetros. Conseguindo medir distancias entre 0 e 255 centímetros com uma precisão de +/- 3 cm. Medindo a distância por calcular o tempo que entre uma onda de som a atingir um objeto e retornar, como o eco. Objetos de grandes dimensões e de superfície dura obtêm melhores leituras, em relação a objetos de fabrico mole, curvo ou pequeno, que podem ser difíceis de detectar.</p>
9842 Servo Motor	
	<p>O servo motor possui um sensor de rotação interna que mede as rotações do motor em graus ou rotações completas, segundo uma acurácia de +/- um grau. Uma rotação equivale a 360 graus.</p>
8529 Cabos conectores	
	<p>O cabo conector permite ligar sensores e motores ao i-Brick. Em cada extremidade possui um conector RJ12.</p>
8529 Cabo USB	
	<p>O cabo conector permite ligar sensores e motores ao I-Brick. Em cada extremidade possui um conector RJ12.</p>



Anexo E: Planificação da prática de ensino supervisionada

Planificação da primeira aula de prática de ensino supervisionada

Primeira aula de Prática de Ensino Supervisionada				Tempo: 90 minutos
Unidade de ensino-aprendizagem: Introdução à programação				
Objetivos de aprendizagem: <ul style="list-style-type: none"> Desenvolver programas no robô através do i-Brick; Interpretar o comportamento de robô. 				Extra: <ul style="list-style-type: none"> Desenvolvimento do pensamento computacional Promoção da atenção
Duração	Atividades	Papel do professor / Papel do aluno	Avaliação / Produtos a desenvolver	Recursos
5 min.	Organização das equipas	À medida que os alunos chegam à sala de aula o professor encaminha-os para o quadro branco onde estão afixados cartões com os elementos de cada equipa e informa os alunos que devem sentar-se junto dos computadores identificados com o número da sua equipa	-	<ul style="list-style-type: none"> Cartões com as equipas
3 min.	Apresentação do professor	O professor apresenta-se à turma e relembra o motivo da sua presença.	-	-
15 min.	Pré Questionário de cariz investigativo	O professor disponibiliza o Pré Questionário na disciplina NXT Heroes do <i>Moodle</i> da escola, e os alunos preenchem-no individualmente.	-	<ul style="list-style-type: none"> Computador Internet <i>Moodle</i> Pré Questionário online
20 min.	Reflexão inicial sobre conceitos de programação	O professor através de questionamento fará uma avaliação diagnóstica sobre conceitos de programação e depois completará o conhecimento dos alunos com auxílio da Apresentação sobre conceitos de programação.	O professor através de observação direta e questionamento avalia as atitudes e comportamentos através dos parâmetros: atenção, sentido de oportunidade, espírito crítico e espírito de iniciativa.	<ul style="list-style-type: none"> Computador Vídeo Projetor Apresentação sobre Conceitos de programação
2 min.	Apresentação da temática	O professor projetará um vídeo sobre a história NXT Heroes	-	<ul style="list-style-type: none"> Computador Vídeo Projetor Vídeo NXT Heroes
40 min.	Desafio 1 – parte 1	O professor disponibilizará na disciplina NXT Heroes do Moodle na escola os slides 1, 2 e 3 do recurso NXT Heroes. Para além disso, o professor fornecerá uma Folha de Testes no I-brick em que os alunos registaram os comandos e interpretação dos mesmos.	<p>A nível de conhecimentos competências e capacidades com alunos serão avaliados pelos seguintes recursos: NXT Heroes e Folha de Testes no i-Brick.</p> <p>A nível de atitudes e comportamentos os alunos serão avaliados segundo os parâmetros:</p>	<ul style="list-style-type: none"> Computador Internet <i>Moodle</i> Envelope Robôs NXT Heroes Cameras de vídeo Folha Testes i-Brick

			interação entre a equipa, espírito de iniciativa e cumprimento das tarefas.	
5 min.	Recolha do Desafio 1	O professor disponibilizará uma Pen drive onde os alunos deverão guardar o Desafio 1.	-	<ul style="list-style-type: none"> • Pen drive • Folha de Testes no i-Brick
	-	-	Ao longo da aula o professor avalia com as atitudes e comportamentos dos alunos aos parâmetros: Pontualidade, Apresentação do material necessário, Respeito, Responsabilidade, Utilização correta dos equipamentos e Respeito pelas normas de funcionamento das salas	-

Planificação da segunda aula de prática de ensino supervisionada

Segunda aula de Prática de Ensino Supervisionada Unidade de ensino-aprendizagem: Introdução à programação			Tempo: 90 minutos	
Objetivos de aprendizagem: <ul style="list-style-type: none"> • Desenvolver programas no robô através do i-Brick; • Interpretar o comportamento do robô; • Compreender conceitos de programação: <ul style="list-style-type: none"> ○ Definir o conceito de programação; ○ Identificar o conceito de programação; ○ Listar os comandos disponíveis no i-Brick; ○ Identificar sensores do robô e suas funcionalidades; ○ Elaborar histórias. 			Extra: <ul style="list-style-type: none"> • Desenvolvimento do pensamento computacional • Promoção da atenção 	
Duração	Estratégias	Atividades	Avaliação / Produtos a desenvolver	Recursos
5 min.	Organização das equipas.	À medida que os alunos chegam à sala de aula o professor relembra os alunos de qual são os lugares reservados para cada equipa. E fala brevemente sobre os objetivos da aula.	-	-
20 min.	Continuação do Desafio 1 – parte 1	O professor fornece aos alunos o feedback escrito e apoia as equipas a melhorar suas soluções. Para além disso, o professor deve certificar-se de que todos os alunos conseguiram desenvolver o seu programa	A nível de conhecimentos competências e capacidades com alunos serão avaliados pelos seguintes recursos: NXT Heroes e Folha de Testes no i-Brick. A nível de atitudes e	<ul style="list-style-type: none"> • Computador • Internet • Moodle • Envelope • Robôs • NXT Heroes • Camaras de vídeo • Folha Testes i-Brick

			comportamentos os alunos serão avaliados segundo os parâmetros: interação entre a equipa, espírito de iniciativa e cumprimento das tarefas.	
20 min.	Desafio 1 – parte 2	O professor disponibiliza os slides 4, 5, 6, 7 do recurso NXT Heroes através da página no <i>Moodle</i> da escola. Os alunos em equipas devem Definir o conceito de programação; Identificar o conceito de programação; Listar os comandos disponíveis no i-Brick; Identificar sensores do robô e suas funcionalidades e Elaborar histórias.	A nível de conhecimentos competências e capacidades com alunos serão avaliados pelo recurso NXT Heroes. A nível de atitudes e comportamentos os alunos serão avaliados segundo os parâmetros: interação entre a equipa, espírito de iniciativa e cumprimento das tarefas.	<ul style="list-style-type: none"> • Computador • Internet • <i>Moodle</i> • Envelope • Robôs • NXT Heroes • Camaras de vídeo
20 min.	Encenação das histórias	O professor organiza as equipas de modo a todas encenarem a sua história com os robôs. Os alunos por sua vez devem designar um elemento para contar a história e outro para interagir com o robô.	-	<ul style="list-style-type: none"> • Robôs
20 min.	Reflexão final sobre conceitos de programação	Em plenário o professor reflete com a turma o que cada uma aprendeu com o Desafio 1.	O professor através de observação direta e questionamento avalia as atitudes e comportamentos através dos parâmetros: atenção, sentido de oportunidade, espírito crítico e espírito de iniciativa.	-
5 min.	Recolha do Desafio 1	O professor disponibilizará uma Pen drive onde os alunos deverão guardar o Desafio 1.	-	<ul style="list-style-type: none"> • Pen drive • Folha de Testes no i-Brick
-	-	-	Ao longo da aula o professor avalia com as atitudes e comportamentos dos alunos aos parâmetros: Pontualidade, Apresentação do material necessário, Respeito, Responsabilidade, Utilização correta dos equipamentos e Respeito pelas normas de funcionamento das salas	-

Planificação da terceira aula de prática de ensino supervisionada

Terceira aula de Prática de Ensino Supervisionada Unidade de ensino-aprendizagem: Introdução à programação				Tempo: 90 minutos
Objetivos de aprendizagem: <ul style="list-style-type: none"> Compreender conceitos de algoritmos; Desenvolver programas em NXT-G utilizando o bloco MOVE; Elaborar pseudocódigo. 				Extra: <ul style="list-style-type: none"> Desenvolvimento do pensamento computacional Promoção da atenção
Duração	Estratégias	Atividades	Avaliação / Produtos a desenvolver	Recursos
15 min.	Questionário sobre o Desafio 1.	O professor disciplina NXT Heroes do <i>Moodle</i> da escola o questionário.	-	<ul style="list-style-type: none"> Questionários Desafio 1 online Computadores Internet
15 min.	Auto e hétero avaliação sobre o Desafio 1.	O professor ouvirá cada uma das equipas fazer a sua auto e hétero avaliação e depois dirá o seu parecer.	-	<ul style="list-style-type: none"> Apresentação da auto e hétero avaliação
15 min.	Reflexão inicial sobre conceitos de algoritmos	<p>O professor através de questionamento fará um paralelo entre o pseudocódigo e a interpretação do comportamento do robô realizada no Desafio 1 e completará o conhecimento dos alunos com auxílio da Apresentação sobre conceitos de algoritmos.</p> <p>Durante a reflexão os alunos realizaram a atividade extra com objetivo de Compreender conceitos de algoritmos</p>	<p>A nível de conhecimentos competências e capacidades com alunos serão avaliados pelo recurso Atividade Extra.</p> <p>A nível de atitudes e comportamentos os alunos serão avaliados segundo os parâmetros: interação entre a equipa, espírito de iniciativa e cumprimento das tarefas.</p>	<ul style="list-style-type: none"> Apresentação sobre Conceitos de algoritmos
10 min.	Proposta do Desafio 2	O professor informará os alunos sobre os objetivos de aprendizagem do desafio. A turma em conjunto definirá a estratégia de resolução do desafio.	-	<ul style="list-style-type: none"> NXT Heroes Cenário
30 min.	Experimentação do Desafio 2	O professor disponibiliza os slides 8 a 13 do recurso NXT Heroes através da página no <i>Moodle</i> da escola. Os alunos em equipas devem Desenvolver programas em NXT-G utilizando o bloco MOVE e Elaborar pseudocódigo para as três fases do desafio.	<p>A nível de conhecimentos competências e capacidades com alunos serão avaliados pelo recurso NXT Heroes.</p> <p>A nível de atitudes e comportamentos os alunos serão avaliados segundo os parâmetros: interação entre a equipa, espírito de iniciativa e cumprimento das tarefas.</p>	<ul style="list-style-type: none"> Computador Internet <i>Moodle</i> Envelope Robôs NXT Heroes Camaras de vídeo Cenário
5 min.	Recolha do Desafio 2 e	O professor disponibilizará uma Pen drive onde os alunos deverão guardar o Desafio 2 e	-	<ul style="list-style-type: none"> Pen drive

	atividade extra	a atividade extra.		
-	-	-	Ao longo da aula o professor avalia com as atitudes e comportamentos dos alunos aos parâmetros: Pontualidade, Apresentação do material necessário, Respeito, Responsabilidade, Utilização correta dos equipamentos e Respeito pelas normas de funcionamento das salas	-

Planificação da quarta aula de prática de ensino supervisionada

Segunda aula de Prática de Ensino Supervisionada Unidade de ensino-aprendizagem: Introdução à programação				Tempo: 90 minutos
Objetivos de aprendizagem: <ul style="list-style-type: none"> Desenvolver programas em NXT-G utilizando o bloco MOVE; Elaborar pseudocódigo. 				Extra: <ul style="list-style-type: none"> Desenvolvimento do pensamento computacional Promoção da atenção
Duração	Estratégias	Atividades	Avaliação / Produtos a desenvolver	Recursos
65 min.	Continuação do Desafio 2	O professor fornece aos alunos o feedback escrito e apoia as equipas a melhorar suas soluções.	<p>A nível de conhecimentos competências e capacidades com alunos serão avaliados pelo recurso NXT Heroes.</p> <p>A nível de atitudes e comportamentos os alunos serão avaliados segundo os parâmetros: interação entre a equipa, espírito de iniciativa e cumprimento das tarefas.</p>	<ul style="list-style-type: none"> Computador Internet Moodle Envelope Robôs NXT Heroes Camaras de vídeo Folha Testes i-Brick
20 min.	Reflexão final sobre conceitos de algoritmos	Em plenário o professor reflete com a turma o que cada uma aprendeu com o Desafio 2.	O professor através de observação direta e questionamento avalia as atitudes e comportamentos através dos parâmetros: atenção, sentido de oportunidade, espírito crítico e espírito de iniciativa.	-
5 min.	Recolha do Desafio 2	O professor disponibilizará uma Pen drive onde os alunos	-	<ul style="list-style-type: none"> Pen drive

		deverão guardar o Desafio 2.		
-	-	-	Ao longo da aula o professor avalia com as atitudes e comportamentos dos alunos aos parâmetros: Pontualidade, Apresentação do material necessário, Respeito, Responsabilidade, Utilização correta dos equipamentos e Respeito pelas normas de funcionamento das salas	-

Planificação da quinta aula de prática de ensino supervisionada

Segunda aula de Prática de Ensino Supervisionada Unidade de ensino-aprendizagem: Introdução à programação				Tempo: 90 minutos
Objetivos de aprendizagem:				Extra: <ul style="list-style-type: none"> Desenvolvimento do pensamento computacional Promoção da atenção
Duração	Estratégias	Atividades	Avaliação / Produtos a desenvolver	Recursos
15 min.	Questionário sobre o Desafio 2.	O professor disponibiliza na disciplina NXT Heroes do <i>Moodle</i> da escola o questionário.	-	<ul style="list-style-type: none"> Questionário Desafio 2 <i>online</i> Computadores Internet
15 min.	Pós Questionário de cariz investigativo	O professor disponibiliza na disciplina NXT Heroes do <i>Moodle</i> da escola o questionário	-	<ul style="list-style-type: none"> Pós Questionário <i>online</i> Computadores Internet
20 min	Reflexão final sobre a intervenção letiva	Em plenário o professor reflete com a turma sobre o que aprenderam com a intervenção letiva.	-	-

Anexo F: Taxonomia dos objetivos de aprendizagem

As taxonomias de objetivos de aprendizagem são utilizadas para fornecer uma linguagem comum para descrever os resultados da aprendizagem e desempenho em avaliações (Fuller, Johnson, Ahoniemi, Cukierman, Hernán-Losada, Jackova, Lahtinen, Lewis, Thompson, Riedeselm & Thompson, 2007). A revisão de literatura revela que a taxonomia com maior impacto nos currículos dos cursos de ciências de computação é a de Bloom na dimensão cognitiva, principalmente pelo facto dos seus níveis serem razoavelmente fáceis de entender e existir literatura de referência que ajuda a usá-la e aplicá-la de modo a ir de encontro às necessidades e exigências das ciências da computação (Fuller et al., 2007). Esta taxonomia tem como desvantagem o facto de os seus níveis serem hierárquicos e ascendentes, ou seja, parte do princípio que a evolução do conhecimento vai do nível mais baixo para o mais alto de modo sequencial (Fuller et al., 2007). Para minimizar esta limitação da taxonomia de Bloom, Fuller e colaboradores (2007) adaptaram esta taxonomia a uma matriz bidimensional, permitindo que os alunos atinjam os níveis superiores de conhecimentos, aplicar e criar, por meio de diferentes trajetórias. Salienta-se que esta matriz foi pensada para a elaboração de objetivos pedagógicos em ciências da computação.

Objetivos pedagógicos Desafio 1 parte 1

Criar		Desenvolver programas diretamente no i-Brick Interpretar o comportamento do robô		
Aplicar				
Nenhum				
	Lembrar	Compreender	Analisar	Avaliar

Para o Desafio 1 parte 1 os objetivos pedagógicos Desenvolver programas diretamente no i-Brick e Interpretar o comportamento do robô, encontram-se no nível Criar que segundo Ferraz e Belhot (2010) significa “criar uma nova solução, estrutura ou modelo utilizando conhecimentos e habilidades previamente adquiridos”, e neste caso os alunos têm que desenvolver soluções próprias, existindo múltiplas possibilidades de respostas, não havendo apenas uma resposta certa. Os mesmos objetivos estão associados igualmente ao nível Compreender porque segundo Ferraz e Belhot (2010) a “construção

de significados caracteriza-se através de linguagem oral, escrita ou gráfica, usando para isto a interpretação, exemplificação, classificação, sumarização, inferência e explicação”, e neste caso está previsto que os alunos aprendam a programar e interpretar o comportamento do robô de uma forma ativa e dinâmica por si só, através da exploração, testes e reflexão. Salienta-se que a interpretação do comportamento do robô é uma introdução à elaboração de algoritmos.

Objetivos pedagógicos Desafio 1 parte 2

Criar			Identificar conceitos de programação	
Aplicar	Listar os comandos disponíveis no i-Brick	Identificar sensores do robô e suas funcionalidades	Elaborar histórias	
Nenhum	Definir o conceito de programação			
	Lembrar	Compreender	Analisar	Avaliar

Enquanto no Desafio 1 parte 2, o objetivo pedagógico Listar os comandos disponíveis no i-Brick encontra-se no nível Aplicar e Lembrar porque os alunos poderão começar por listar os comandos que conhecem, mas para os outros de interagir com o robô de modo a conhece-los, constitui-se assim uma atividade prática semi orientada. O objetivo pedagógico Identificar sensores do robô e suas funcionalidades encontra-se no nível Aplicar e Conhecer, porque com base no conhecimento adquirido através da experimentação do robô. O objetivo pedagógico elaborar histórias, foi de encontro com Rusk, Resnick, Berg e Pezalla-Granlund (2008), que defendem que é vantajoso serem os próprios alunos a desenvolverem os seus próprios contextos de aprendizagem. Neste caso, como os alunos não tinham experiência de programação e não conheciam ainda as funcionalidades do robô, optou-se por deixar os alunos explorar as potencialidades do robô através da experimentação da programação e reflexão sobre o mesmo e depois face ao produto final os alunos constroem o seu contexto de aprendizagem, ou seja, contextualizam através de uma pequena história o comportamento do robô. Desta forma, os programas passam a ser personalizados pelos alunos e mais originais, sendo este objetivo pedagógico uma introdução à capacidade dos alunos de pensar, identificar e elaborar problemas; capacidade imprescindível tanto para a programação como para outras áreas do saber.

Objetivos atividade extra

Criar				
Aplicar				
Nenhum				Compreender conceitos de algoritmos
	Lembrar	Compreender	Analisar	Avaliar

Durante a reflexão inicial de conceitos de algoritmos foi promovida uma atividade extra, que embora estivesse relacionada com o Desafio 1, não fez parte integrante deste, com o objetivo pedagógico: Compreender conceitos de algoritmos. Esta atividade pretendeu, com base no programa elaborado em equipa fazer os alunos refletir sobre as características dos algoritmos, e avaliar se a interpretação do comportamento do robô que fizeram poderia ser considerada um algoritmo ou não. Para a interpretação do comportamento do robô ser considerada um algoritmo, teria que ser rigoroso, eficaz e finito.

Objetivos pedagógicos Desafio 2

Criar		Desenvolver programas através do NXT-G aplicando o bloco MOVE Elaborar pseudocódigo		
Aplicar				
Nenhum				
	Lembrar	Compreender	Analisar	Avaliar

O Desafio 2 teve como finalidade proporcionar aos alunos um contato mais abrangente com a multidisciplinaridade da robótica e da programação. Para além da disciplina de Português através da elaboração de histórias, foram incluídos conceitos das disciplinas preferidas dos alunos Matemática e Física, concretizados nos recursos: fita métrica para realizarem conversões e saco de pedras para equilibrar o peso do robô. No entanto o foco do desafio continua a ser a aprendizagem de programação.

Depois de ultrapassado o Desafio 1 e estando agora os alunos familiarizados com os componentes e funcionalidades gerais do robô, tornou-se necessário aumentar o grau de complexidade do Desafio 2, passando-se assim à programação em NXT-G que permite um maior controlo do movimento do robô, através do bloco MOVE. O objetivo de aprendizagem Desenvolver programas em NXT-G utilizando o bloco MOVE encontra-se no nível Criar porque os alunos elaboram soluções únicas, e no nível Compreender aprendem a fazê-lo na prática por experimentação. Ao ser introduzido o conceito de algoritmo, através da atividade extra, neste desafio pretende-se que os alunos deixem de interpretar o comportamento de robô e passem a elaborar pseudocódigo, respeitando as suas características de rigor, eficácia e finitude. Assim, o objetivo de aprendizagem Elaborar pseudocódigo encontra-se nos níveis Criar e Compreender porque se desenvolve em paralelo com o objetivo anterior.

Anexo G: Critérios de avaliação da disciplina Aplicações Informáticas B

CRITÉRIOS DE AVALIAÇÃO DA DISCIPLINA DE
APLICAÇÕES INFORMÁTICAS B

Conhecimentos	Testes sumativos e/ou Trabalhos Individuais	45%	90%
Competências		Trabalhos escritos e/ou orais, em grupo	
Capacidades			
Atitudes e Comportamentos	Pontualidade, apresentação do material necessário, atenção nas aulas, sentido de oportunidade, espírito crítico, interação, espírito de iniciativa, cumprimento das tarefas, respeito, responsabilidade, utilização correta dos equipamentos e respeito pelas normas de funcionamento das salas.		10%

✂ _____

_____, Encarregado de Educação
do aluno _____ do 12º Ano, Turma
____, Nº____, tomei conhecimento dos critérios de avaliação da disciplina de Aplicações
Informáticas B.

O Encarregado de Educação:

2012 / 09 / ____

Anexo H: Grelhas das Atitudes e comportamentos

Reflexões				
	Atenção nas aulas	Sentido de oportunidade	Espírito crítico	Espírito de iniciativa
Aluno 1				
Aluno 2				
....				
Aluno N				

Desafios				
	Atenção nas aulas	Interação entre a equipa e o professor	Espírito de iniciativa	Cumprimento das tarefas
Aluno 1				
Aluno 2				
.....				
Aluno N				

Comportamento					
	Pontualidade	Respeito	Responsabilidade	Utilização correta dos equipamentos	Respeito pelas normas de funcionamento das salas
Aluno 1					
Aluno 2					
....					
Aluno N					

Anexo I: Regras de avaliação

Com a finalidade de reduzir a ambiguidade na atribuição de classificações, foram criadas regras para atribuição de classificação para os parâmetros: programação, interpretação e reflexão. Para o parâmetro programação, mais especificamente para o registo de vídeo os alunos teriam apenas 5% pela submissão do mesmo, e +20% se este refletisse o comportamento do robô descrito na interpretação do comportamento do robô e ser o resultado do programa submetido. Quanto ao uso de sensores os alunos obteriam 5% caso no seu programa optassem por usar outros tipos de comandos em vez de sensores, ou 10% se usassem apenas um sensor, ou 15% se no mesmo programa usassem dois sensores iguais ou 25% caso usassem dois sensores diferentes. Salienta-se o facto, de que na programação direta no i-Brick em cada programa apenas é possível utilizar no máximo dois sensores, especificamente no segundo e quarto comandos disponíveis. Em relação ao não uso de empty's, esta regra foi criada visto se pedir aos alunos que criassem programas de cinco comandos, se estes escolhessem em qualquer um dos passos "empty", o programa seria considerado como tendo mesmo de cinco passos contrariamente ao pedido. Assim, os alunos que não usassem empty's teriam 25% da classificação e caso usassem empty's independente do número de vezes obteriam 0% de classificação. Em termos de originalidade, aqui se pretendia que os alunos apesar de criarem programas individualmente, tivessem alguma noção dos programas da sua equipa, e tendo em vista uma maior profundidade de conhecimento, apelando ao facto de os alunos aprenderem uns com os outros, e não havendo a possibilidade de um aluno explorar todos os sensores, visto serem quatro no total e os alunos em cada programa apenas poderem explorar no máximo dois. Assim, um programa foi considerado mais ou menos original se fizessem sentido +5%, e caso tivesse usado um sensor teria +5% e se esse sensor fosse diferente dos utilizados pela sua equipa teria um bónus de +5%, por outro lado caso tivessem usado dois sensores +10% e se esses dois sensores tivessem sido combinados de forma diferente dos elementos da sua equipa teria um bónus de +10%.

Para o parâmetro interpretação do comportamento do robô, cada uma das cinco interpretações valeria 20%, e caso alguma estivesse incompleta valeria apenas 10%. Em relação ao parâmetro reflexão, mais especificamente aos conceitos de programação os alunos teriam +5% se referissem que um programa é um conjunto de instruções, +5% com objetivo, +5% numa linguagem de programação específica e teriam +5% se incluíssem outros dados relevantes à definição. Enquanto na aplicação de conceitos de programação, os alunos teriam +10% se referissem o conjunto de instruções que

utilizaram, +10% se especificassem o objetivo do seu programa, +5% se indicassem a linguagem de programação que utilizaram e +5% se incluíssem outros dados relevantes à sua experimentação da programação com o robô. Para a especificação de como se programa no i-Brick os alunos obteriam +4% se especificassem que para programar o robô usaram o Menu NXT Program do i-Brick, +4% se referissem que no máximo poderiam criar programas com cinco instruções e +2% por outras informações válidas. Na identificação das funcionalidades do i-Brick, visto tratar-se de uma simples listagem os alunos obteriam 10% se esta estivesse correta e 0% caso estivesse incompleta ou incorreta. Na identificação dos sensores os alunos obteriam +5% por identificarem corretamente os sensores que utilizaram o programa e +15% por explicarem corretamente a sua atuação. Por último na criação da história, os alunos obteriam 5% se a história apenas proporcionasse um contexto ao programa, ou 8% se a história contextualiza-se o programa e inclui-se uma explicação de alguns comandos ou 10% se a história contextualiza-se contexto e inclui-se uma explicação clara de todos os comandos.

Regras para classificação do Desafio 1

Programação

Registo de vídeo (25%) = submissão (5%) + claro reflexo do programa e interpretação (20%)

Uso de sensores (25%) = utilização de outros comandos (5%) || utilização de um sensor (10%) || utilização de dois sensores iguais (15%) || utilização de dois sensores iguais (25%)

Não uso de empty's (25%) = uso de empty's (0%) || não uso de empty's (25%)

Originalidade (25%) = programa faz sentido (5%) + [(usado um sensor (5%) + bônus (5%)) || (usado dois sensores (10%) + bônus (10%))]

Interpretação

Interpretação passo N (20%) = interpretação incompleta (10%) || interpretação correta (20%)

Reflexão

Conceitos de programação (20%) = conjunto de instruções (5%) + com objetivo (5%) + numa linguagem de programação (5%) + extra (5%)

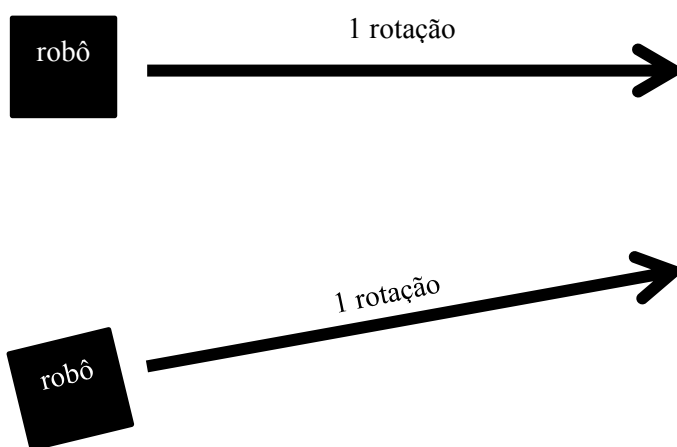
Aplicações de conceitos de programação (30%) = conjunto de instruções (10%) + com objetivo (10%) + numa linguagem de programação (5%) + extra (5%)

Identificação das funcionalidades do i-Brick (10%) = 10% correta || 0% incompleta/incorreta

Identificação dos sensores (20%) = +5% identificação de sensores +15% explicação de sua atuação

História (10%) = 5% tem contexto || 8% tem contexto e explica alguns comandos || 10% tem contexto e explica de forma clara de todos os comandos

A avaliação do pseudocódigo estava dependente do número de bloco utilizados e da correta especificação de cada um. Enquanto a avaliação dos programas dependeu do número de bloco utilizados e a correta utilização do bloco MOVE segundo uma duração específica. O registro de vídeo é considerado correto se realizasse as tarefas descritas no algoritmo e concretizadas pelo programa. A importância do registro de vídeo deve-se a que neste desafio, o manuseamento do robô é um fator essencial para o sucesso dos programas. Por exemplo, ao configurar o robô para se mover em frente durante uma rotação, a precisão do movimento depende de iniciar o programa de forma certa, ou seja, se quisermos que o robô ande em frente mas o posicionarmos enviesado, o robô ao seguir em frente tal como programado, vai enviesar no sentido inclinado.



Movimento do robô

Anexo J: Exemplo de aplicação das regras de avaliação

Desafio 1

PROGRAMAÇÃO

	Registo de vídeo	Uso de sensores	Não usou emptys	Originalidade	Total
Equipa 1	25%	25%	25%	25%	100%
Aluno 1	25%	25%	25%	20% a)	75%
Aluno 2	25%	15%	25%	15% b)	65%
Aluno 3	25%	15%	25%	15% c)	65%
Aluno 4	25%	25%	25%	20% d)	75%

OBSERVAÇÕES

a) O programa faz sentido +5, usaram 2 sensores diferentes +5+5, mas um dos sensores é usados por outros elemento da equipa +5

b) O programa faz sentido +5, usaram 2 sensores diferentes +5+5, no entanto estes sensores também são usados por outros membros da equipa

c) O programa faz sentido +5, usaram 2 sensores diferentes +5+5, no entanto estes sensores também são usados por outros membros da equipa

d) O programa faz sentido +5, usaram 2 sensores diferentes +5+5, mas um dos sensores é usados por outros elemento da equipa +5

INTERPRETAÇÃO

	Interpretação Passo 1	Interpretação Passo 2	Interpretação Passo 3	Interpretação Passo 4	Interpretação Passo 5	Total
Equipa 1	20%	20%	20%	20%	20%	100%
Aluno 1	20%	20%	20%	20%	10% e)	80%
Aluno 2	20%	20%	20%	20%	20%	100%
Aluno 3	20%	20%	20%	20%	20%	100%
Aluno 4	20%	20%	20%	20%	20%	100%

OBSERVAÇÕES

e) Interpretação do passo 5 incompleta +10

REFLEXÃO

	Definição de programação	Aplicação de Conceitos de programação	Como se programa no brick	Identificação das funcionalidades do brick	Identificação dos sensores e sua atuação	História (interpretação)	Total
	20%	30%	10%	10%	20%	10%	100%
Equipa 1	13% f)	30%	7% g)	10%	20%	10%	70%

OBSERVAÇÕES

f) Não é especificado que um programa é feito com um objetivo. Demasiada informação extra e pouca objetividade na definição em si.

g) Não são especificados os nomes dos Menus. Escrita confusa.

Anexo K: Disciplina NXT HEROES - Moodle

Durante a observação das aulas da professora cooperante foi notado que os alunos estavam familiarizados com a utilização da plataforma *Moodle*. A Escola Secundária de Camões acedeu ao pedido de criação da disciplina NXT Heroes e esta seguiu uma formatação por tópicos: Semana de treino, Desafio 1 e Desafio 2. O *Moodle* serviu apenas para disponibilizar recursos aos alunos. Não foram feitas submissões pela plataforma porque os trabalhos dos alunos iriam exceder o tamanho permitido definido pela Escola para submissões, para esse efeito foi usado uma Pen Drive. De modo a não sobrecarregar o servidor da Escola Secundária de Camões, todos os elementos inseridos desde imagens, vídeos e documentos foram alojados em servidores públicos grátis, nomeadamente o Youtube, DropBox e Photobucket.

1

Semana de Treino

sob orientação das professoras Diana Oliveira e Mónica Batista



Será que os habitantes de Smalville podem ter esperança?

2

Desafio 1

Introdução à Programação
Conceitos de Programação



- Questionário NXT Heroes (1)
- Conceitos de Programação
- Desafio 1 - parte 1
- Robô - Basquetebolista
- Robô - Tribot
- Robô - Spike
- Desafio 1 - parte 2

3

Desafio 2

Introdução à Programação
Conceitos de Algoritmos

- Conceitos de Algoritmos
- Formulário de reflexão sobre o conceito de algoritmo
- Desafio 2
- Questionário NXT Heroes (2)

Anexo L: Apresentação sobre conceitos de programação - preenchido





Um programa é...
... um conjunto de:

- INSTRUÇÕES
- ORDENS
- COMANDOS


escritos numa **LINGUAGEM** de **PROGRAMAÇÃO** específica,
como por exemplo:

- Java
- C
- C++
- NXT-G


com o fim de realizar uma determinada **TAREFAS**.

A programação é...
...o acto de programar.

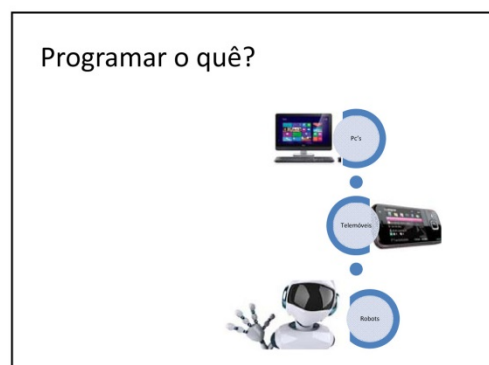


ARTE


+




CIÊNCIA




Programar para quê? Para criar....






Jogos



Páginas Web



Histórias

Bibliografia

- Aguiar, L. (2008). Fundamentos de programação: algoritmos, estruturas de dados e objectos (P. H. Costa do Valle, Trad.). São Paulo: McGraw-Hill (Obra original publicada em 2003).
- Gomes, A., Henriques, J., Mendes, A. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação e Tecnologia*, 1, 93-103.

Anexo M: Apresentação sobre conceitos de algoritmos - preenchido



Definição de Algoritmo

Informalmente, um algoritmo é sequência de instruções que podem ser executadas de modo a atingir um determinado objetivo, sendo que este objetivo, está sempre associado à solução de um problema.

Aplicações Informáticas B

Algoritmos no quotidiano

Um algoritmo não representa, necessariamente, um programa de computador, mas sim os passos necessários para realizar uma tarefa a ser executada por um agente, o qual pode ser humano, mecânico, electrónico, ou outro.

Aplicações Informáticas B

Caraterísticas dos Algoritmos

- Ser **rigoroso**, no sentido de cada instrução ser simples e sem ambiguidade, especificando exacta e rigorosamente o que deve ser feito, sem margem para segundas interpretações.
- Ser **eficaz**. Cada instrução do algoritmo deve ser suficientemente básica e bem compreendida, de modo resolver um problema ou cumprir o objetivo pretendido.
- Ser **finito**, ou seja, o algoritmo deve levar a uma situação em que não existam mais instruções a ser executadas.

Descrição dos Algoritmos

Existem três formas de descrever algoritmos:

- Narrativas;
- Pseudolinguagens ou pseudocódigo;
- Linguagens gráficas.

A pseudo-linguagem é uma linguagem de especificação de algoritmos, que consiste na escrita de instruções através de palavras semelhantes com o Inglês ou Português, que facilitam tanto a escrita como a leitura de programas.

Utilidade do Pseudocódigo

Inicio do loop
 Espera até que o Sensor de Toque seja pressionado
 Se Sensor de Luz > 42 então
 Usa o block de Som para dizer Red
 Senão
 Usa o block de Som para dizer Blue
 Fim se
 loop ilimitado

Aplicações Informáticas B

Atividade de Reflexão

Tendo em conta o programa criado em equipa, reflita se corresponde efetivamente a um algoritmo.

Aplicações Informáticas B

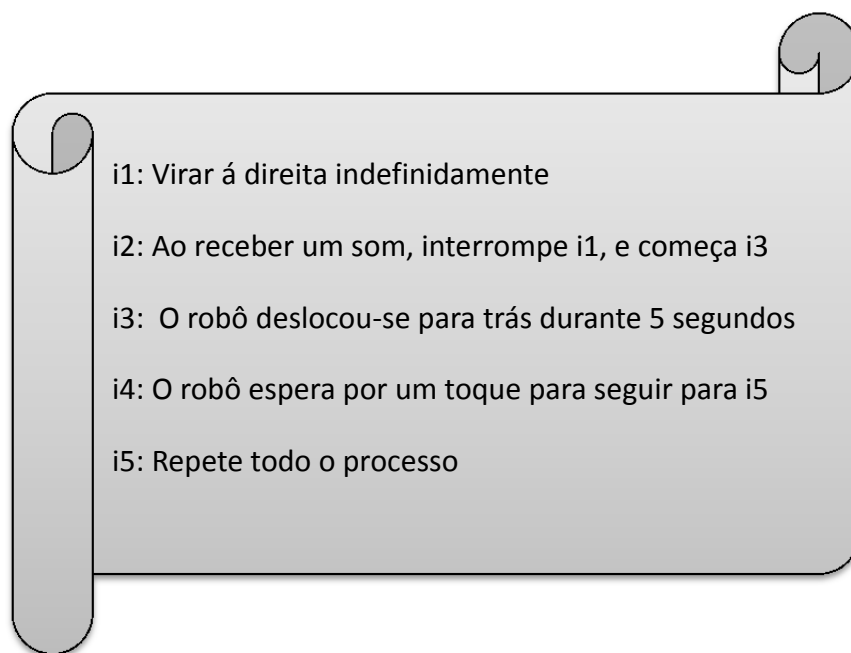
Referências

Martins, J., & Cravo, M. (2011). *Fundamentos da Programação utilizando múltiplos paradigmas*. Lisboa: Ensino da Ciência e da Tecnologia.

Cormen, T., Leiserson, C., Rivest, R., & Stein (2009) *Introduction to Algorithms*, Third edition. The MIT Press.

Anexo N: Exemplo de Folha de Reflexão de algoritmos

EQUIPA 1



Tendo em consideração as características dos algoritmos, reflitam sobre:

- **Rigoroso.** O algoritmo é simples e sem ambiguidades?

Rigoroso	Justificação
Sim <input type="checkbox"/>	
Não <input type="checkbox"/>	

- **Eficaz.** O algoritmo cumpre o objetivo pretendido?

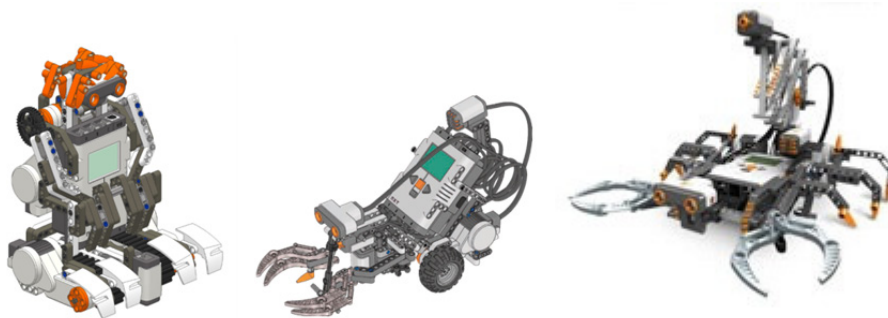
Eficaz	Justificação
Sim <input type="checkbox"/>	
Não <input type="checkbox"/>	

- **Finito.** O algoritmo chega ao fim?

Finito	Justificação
Sim <input type="checkbox"/>	
Não <input type="checkbox"/>	

Anexo O: Modelos de robôs da intervenção letiva

Para a concretização dos desafios foram utilizados 6 *kits* da Lego® Mindstorms® Education, um para cada equipa de alunos. Sendo a duração da prática de ensino supervisionada de apenas cinco aulas de noventa minutos cada, a construção do robô não foi incluída na planificação da mesma. Antes foram construídos previamente dois robôs de cada um dos três modelos. Visto no seguimento dos desafios os robôs teriam que evidenciar uma habilidade especial, mas sem uso de sensores, a escolha dos modelos utilizados recaiu sobre o facto de possuíram três motores. Onde dois motores seriam usados para controlar o movimento do robô e o terceiro serviria como a habilidade especial. Foram escolhidos os modelos do *site* oficial da Lego® Mindstorms®: Basquetebolista que lança bolas, Spike que mexe a cauda e o Tribot que consegue agarrar objetos. O sorteio dos robôs foi realizado através de envelopes (Anexo Q).



Robôs usados na prática de ensino supervisionada

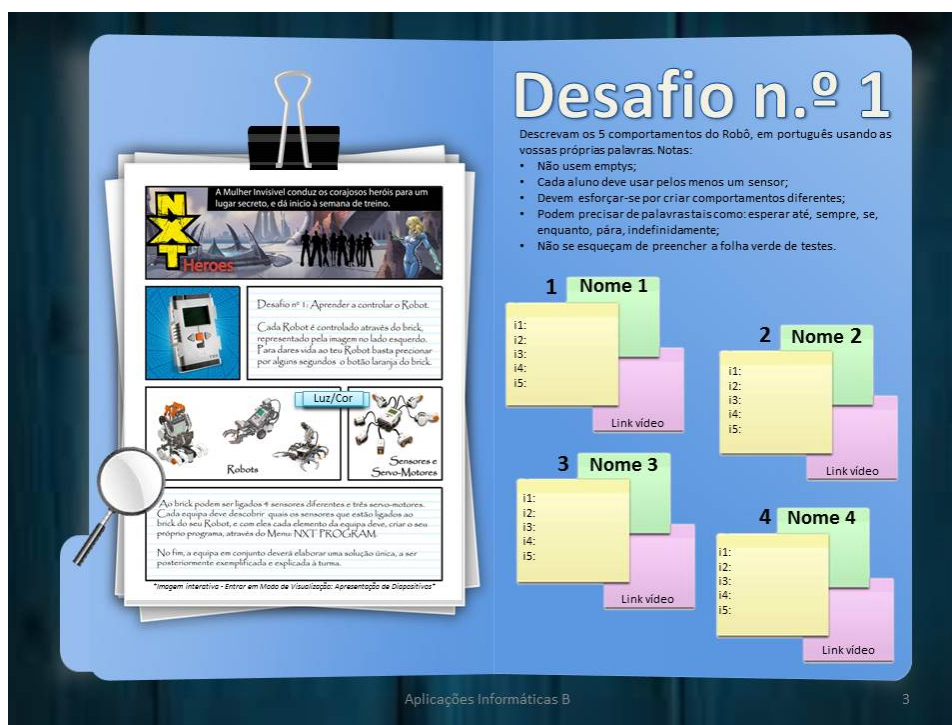
Anexo P: Desafio NXT HEROES

Para auxiliar os alunos na concretização dos desafios e evidenciar o cumprimento dos objetivos de aprendizagem delineados, foi elaborado o denominado NXT Heroes, onde os alunos documentariam o desenrolar dos desafios. No slide 2 os alunos preencheriam o perfil da equipa, por especificar os seus elementos, o robô com que trabalharam e a sua característica encontrada no papel amarelo do envelope.



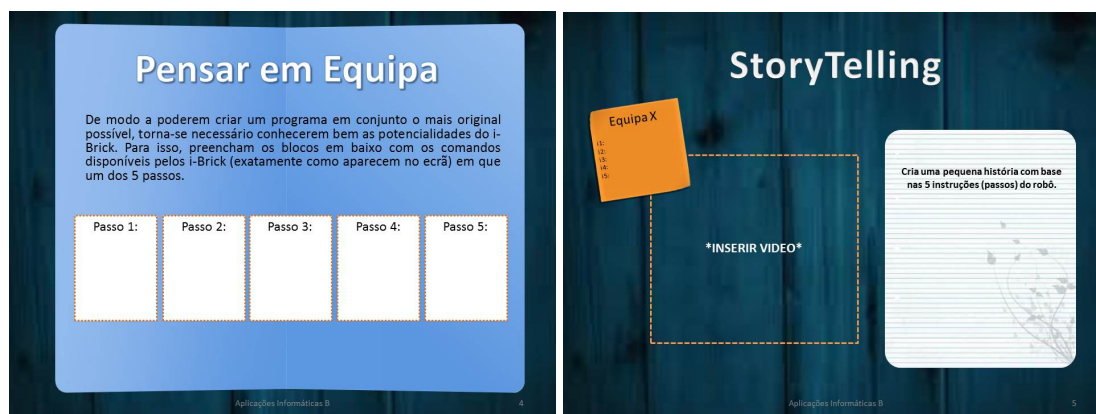
Slides 1 e 2

O slide 3 continha a história em banda desenhada do Desafio 1, tal como algumas das regras de avaliação referentes à programação do robô. Neste slide, cada elemento da equipa registaria a interpretação do comportamento do robô correspondente ao seu programa e criaria um *hiperlink* para o registo de vídeo.



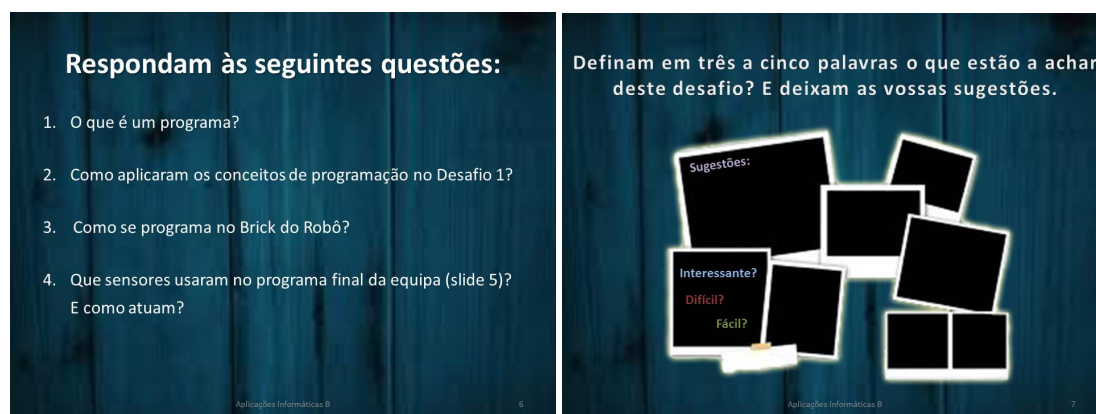
Slide 3

No slide 4, os alunos registrariam a listagem de comandos disponíveis no i-Brick. Enquanto no slide 5, os alunos deveriam descrever o programa escolhido para representar a equipa, registrar a história elaborada sobre esse programa e inserir um vídeo exemplificativo do robô a viver a história.



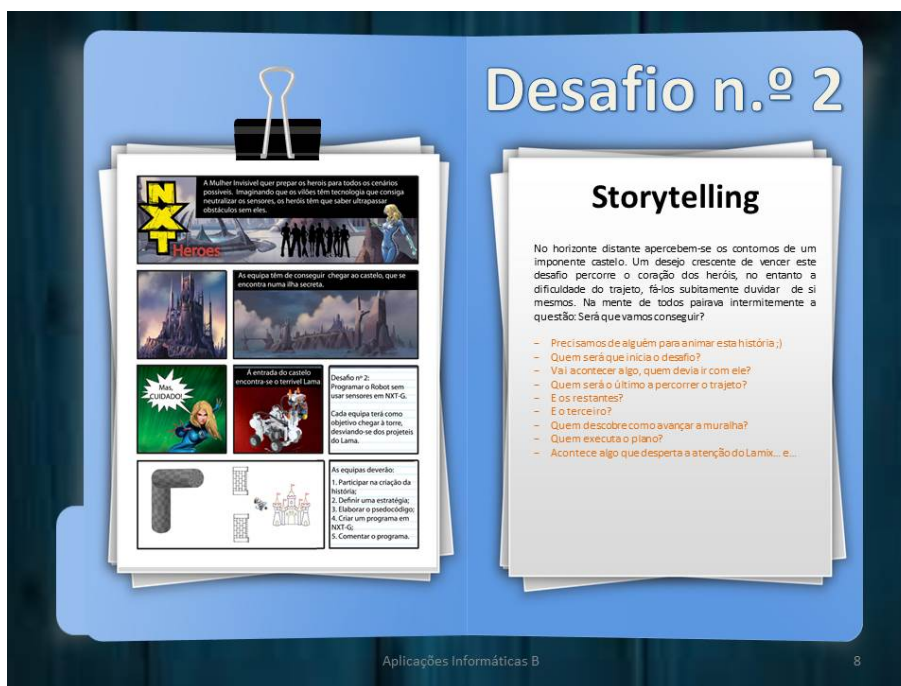
Slides 4 e 5

No slide 6, os alunos deveriam registrar a sua reflexão em equipa sobre as questões: O que é um programa? Como aplicaram os conceitos de programação no Desafio 1? Que sensores usaram no programa final da equipa? E como atuam? O slide 7 serviria para os alunos descreverem o desafio experienciado com cinco palavras, ou para apontamento de sugestões.



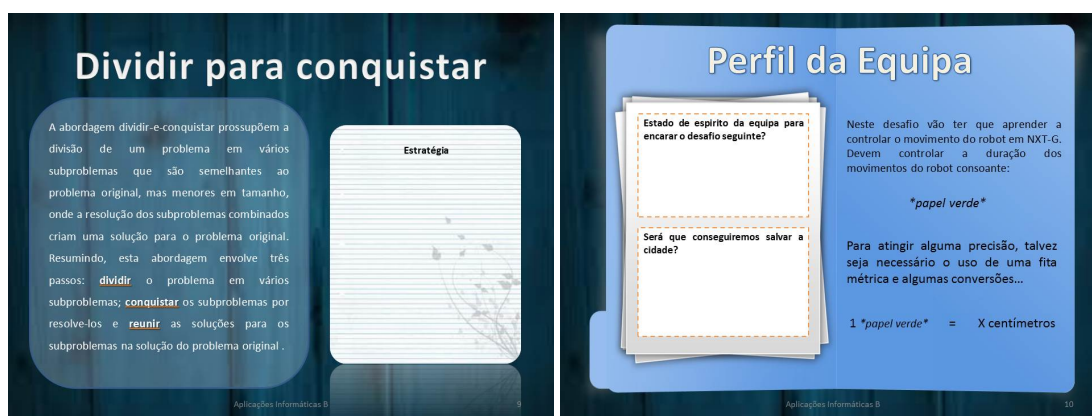
Slides 6 e 7

Para o Desafio 2, no slide 8 foi fornecida uma história em banda desenhada introdutória do tema, e um guião de *storytelling* onde as equipas através das características contidas dos seus envelopes, criariam em conjunto o seu próprio contexto de aprendizagem.



Slide 8

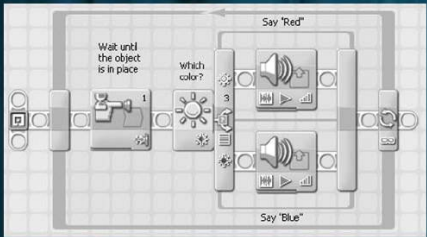
O slide 9, introduz os alunos à abordagem dividir para conquistar. Pretende-se que as equipas em conjunto depois de definirem a estratégia geral para a concretização do desafio, dividindo-o em fases, registem-na. Depois através do slide 10, cada equipa deverá preencher o seu perfil indicando o estado de espírito da equipa em relação ao desafio e ao seu sucesso Para além disso, devem registar a forma como iriam controlar a Duração do robô, traduzida pelo papel verde do envelope. Por último é feita a sugestão de os alunos usarem a fita métrica para calcular com mais precisão a duração do movimento do robô.



Slides 9 e 10

No slide 11 (Figura 25), é fornecido aos alunos um programa em NXT-G exemplo e respetivo pseudocódigo, baseado no livro de Griffin (2011) *The art of LEGO® MINDSTORMS® NXT-G programming*.

Programa Exemplo



Pseudocódigo:

Início do loop
 Espera até que o **Sensor de Toque** seja pressionado
 Se **Sensor de Luz** > 42 então
 Usa o **block de Som** para dizer Red
 Senão
 Usa o **block de Som** para dizer Blue
 Fim Se
 loop ilimitado

Griffin, T. (2010). *The art of LEGO® MINDSTORMS® NXT-G programming*.
 San Francisco: No Starch Press.

Aplicações Informáticas B 11

Slide 11

Em relação os slides 11 e 12 (Figura 26), estes sevem para as equipas em cada fase registarem quem fez de programador e ajudante, com intuito de fazer os alunos rodarem de papéis. Tal como deveriam registar uma imagem comprovativa no programa criado em NXT-G e o respetivo pseudocódigo. Por último o slide 13, é igual ao slide 7 e serviria para os alunos descreverem o desafio experienciado com cinco palavras, ou para apontamento de sugestões.

Fase N

Quem são os programadores?

Quem são os que irão testes os robots no cenário e tirar medidas?

Aplicações Informáticas B 12

Fase N

Pseudocódigo

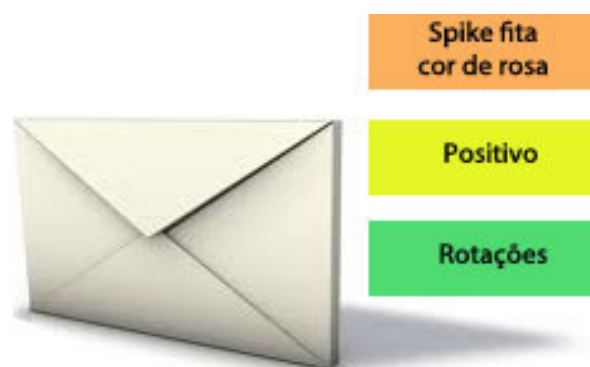
Imagem do programa em NXT-G

Aplicações Informáticas B 13

Slides 12 e 13

Anexo Q: Envelopes

Foram organizados seis envelopes um para cada equipa para sortear robôs, característica e modo de controlar o robô. Os robôs disponíveis, correspondentes à tira de papel laranja eram: Tribot fita branca, Tribot fita azul escura, Spike fita rosa, Spike fita azul claro, Basquetebolista fita vermelha e Basquetebolista fita verde. As características dos robôs seriam usadas na elaboração do problema para o Desafio 2, correspondentes à tira de papel amarela eram: Positivo, Médico, Forte, Assustadiço, Corajoso e Génio. Por último e para referente do Desafio 2, os modos de controlar o robô, correspondentes à tira de papel verde eram: ilimitado, graus, rotações e tempo. O envelope A continha os seguintes atributos: Tribot fita branca, Médico e Rotações. O envelope B continha os seguintes atributos: Tribot fita azul escura, Génio e Graus. O envelope C continha os seguintes atributos: Spike fita rosa, Positivo e Rotações. O envelope D continha os seguintes atributos: Spike fita azul clara, Corajoso e Tempo. O envelope E continha os seguintes atributos: Basquetebolista fita vermelha, Forte e Graus. O envelope F continha os seguintes atributos: Basquetebolista fita verde, Assustadiço e Ilimitado.



Exemplo do conteúdo de um envelope

**Anexo R: Folha de Testes do i-Brick,
comparação entre programação no i-Brick e NXT-G**

TESTES no i-BRiCK

Equipa: _____

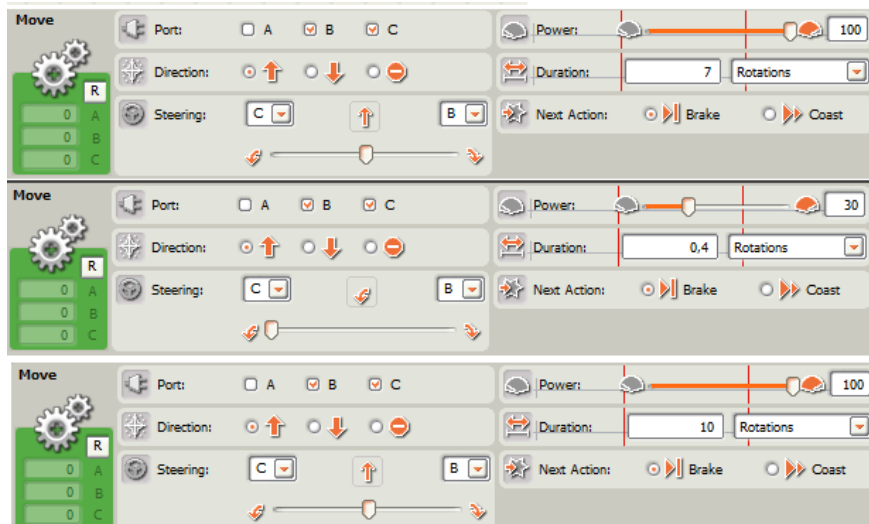
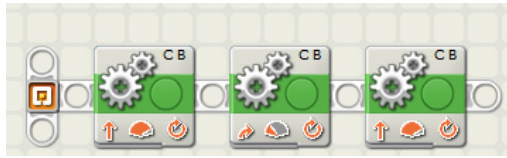
Programador: _____

Ajudante: _____

	Comando brick	Interpretação do comando
1	Forward 5	O robô desloca-se para a frente 5 rotações
2	Sound	O robô espera até ouvir um som
3	Backright 2	O robô vai para trás enquanto vira para a direita 2 rotações
4	Wait 2	O robô espera 2 segundos para seguir para o próximo passo
5	Loop	O robô repete todas as instruções anteriores

	Comando brick	Interpretação do comando
1		
2		
3		
4		
5		

Exemplo de programa em NXT-G correspondente ao Subir a rampa do cenário



Pseudocódigo

O robô dá início ao movimento com direção em frente com os motores B e C, durante 7 rotações com power 100;

O robô muda de direção para a esquerda com os motores B e C, durante 0,4 rotações com power 30;

O robô continua o movimento em frente com os motores B e C durante 10 rotações com power 100.

Anexo S: Pré Questionário Atitudes e Percepções

Questionário sobre atitudes e percepções e atenção (1)

A professora Diana Oliveira no âmbito da Prática de Ensino Supervisionada integrante do Mestrado de Ensino de Informática da Universidade de Lisboa, está a desenvolver um estudo sobre o impacto da robótica educativa no ensino e aprendizagem de programação à disciplina de Aplicações Informáticas B, mais especificamente aos conteúdos: conceitos de programação e algoritmos. Este questionário constitui-se parte fundamental do estudo e destina-se a recolher informação sobre as atitudes e percepções dos alunos em relação a aprender programação com robôs. O questionário leva aproximadamente 5 minutos a responder. O preenchimento é individual e as respostas de cada inquirido serão salvaguardadas por confidencialidade, servindo apenas para fundamentação empírica do estudo. Desde já agradeço a colaboração. Nota: entenda-se que no questionário a palavra "ansioso" é usada não como sinónimo de "estar nervoso", mas como "desejar veemente".

*1. Primeiro e último nome

*2. Atitudes e Percepções

	Não concordo totalmente	Não concordo parcialmente	Indiferente	Concordo parcialmente	Concordo totalmente
Estou ansioso para aprender a programar.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estou ansioso para aprender a programar com robôs.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Concluído

[Ativados pela SurveyMonkey](#)
[Crie seus próprios questionários online gratuitos agora!](#)

Anexo T: Exemplo de feedback escrito

Equipa 2

Aluno R

Bom trabalho, todos os comandos e a respetiva interpretação, estão corretos. No entanto, não te esqueças das regras do slide 3, entre ela encontra-se esta: " Cada aluno deve usar pelos menos um sensor", no teu programa repetiste duas vezes *Wait* (que não é um sensor), aproveita um desses momentos e inclui pelo menos um sensor, e assim o programa em si ficará mais interessante.

Aluno P

Para enriquecer o teu programa em vez de explorares duas vezes o mesmo sensor, experimenta dois diferentes. Por outro lado gostava que explicasses melhor no comando BackRight. **Ou seja quantas rotações ele faz até passar para comando seguinte?** Por último na interpretação do comando LOOP, o que significa andar à roda? Explica melhor este comando. Apesar destas notas saliento que tens um programa com grande potencial.

Aluno T

Os nomes dos comandos e respetiva interpretação estão corretos, no entanto o programa é incoerente, porque na terceira instrução do i-brick não é possível escolher SOUND, tal como na quarta não é possível escolher FORWARD 5. Por favor, corrige estas situações, de resto não tenho mais nada a apontar. Continuação de bom trabalho.

Anexo U: Questionário Desafio 1 e análise

Questionário de Equipas sobre o Desafio 1

* 1. Número da Equipa?

- ☐ 1
☐ 2
☐ 3
☐ 4
☐ 5
☐ 6

* 2. Opinião da Equipa sobre o Desafio 1

	Não concordo totalmente	Não concordo parcialmente	Indiferentes	Concordo parcialmente	Concordo totalmente
Gostei de programar o robot no brick	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Achei difícil programar o robot no brick	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gostei de interpretar o comportamento do robot	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Achei difícil interpretar o comportamento do robot	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gostei de criar a história representativa do comportamento do robot	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Achei difícil criar a história representativa do comportamento do robot	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gostei do desafio 1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Achei difícil o desafio 1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Considero que aprendi Conceitos de Programação com o desafio 1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

* 3. Comentário Geral / Sugestões

No início da terceira aula foi aplicado um questionário às equipas de alunos sobre as suas perspetivas em relação ao Desafio 1, composto pelas três dimensões: preferências, dificuldades e aprendizagens. Em relação à dimensão de preferências das equipas, as médias mais elevadas 3,83, revelam que as equipas gostaram mais foi do Desafio 1 e de programar o robô diretamente no i-Brick. Mas salienta-se o facto de o desvio padrão em relação ao gosto pelo Desafio 1 de 0,41 ser menor que o desvio padrão do gosto de programar diretamente o robô com 0,75, denotando que houve maior concordância de respostas no gosto pelo Desafio 1. Seguidamente a média do gosto pela interpretação do comportamento do robô de 3,67 denota que os gostaram de fazê-lo. O que não gostaram tanto foi criar histórias, com média de 2,67, podendo estar relacionado com o facto de no questionário de caracterização da turma os alunos terem afirmado que a disciplina de que gostavam menos era Português.

Dimensão preferências em relação ao Desafio 1

	Média	Desvio padrão
Gostei de programar o robô diretamente no i-Brick	3,83	0,75
Gostei de interpretar o comportamento do robô	3,67	1,21
Gostei de criar a história representativa do comportamento do robô	2,67	1,03
Gostei do desafio 1	3,83	0,41

Em relação à dimensão dificuldade os resultados revelam que as equipas foram unânimes, evidenciado pelo valor 0 de desvio padrão, em afirmar que não acharam difícil programar o robô no i-Brick, refletido da média de 1. Seguidamente, as equipas não consideraram o Desafio 1 em si difícil, com média de 1,17. Tal como, não consideraram muito difícil interpretar o comportamento do robô e elaborar a correspondente história, com médias correspondentes de 1,67 e 1,83.

Dimensão dificuldades em relação ao Desafio 1

	Média	Desvio padrão
Achei difícil programar o robô no i-Brick	1,00	0
Achei difícil interpretar o comportamento do robô	1,67	0,52
Achei difícil criar a história representativa do comportamento do robô	1,83	0,98
Achei difícil o Desafio 1	1,17	1,22

Nota: Escala: 1 não concordo totalmente, 2 não concordo parcialmente, 3 indiferente, 4 concordo parcialmente e 5 concordo totalmente.

Por último em relação à dimensão aprendizagens, as equipas afirmam que conseguiram aprender alguns conceitos de programação com o Desafio 1, traduzido na média de 3,17.

Dimensão aprendizagens em relação ao Desafio 1

	Média	Desvio padrão
Considero que aprendi conceitos de programação com o desafio	3,17	1,17

Em relação à questão aberta de comentário ou sugestão sobre o desafio, dois foram categorizados como positivos, três como sugestivos e uma como neutro. Nos comentários positivos os alunos qualificam a experimentação do Desafio 1 como “giro” e “interessante”. Enquanto nos comentários sugestivos, os alunos referem necessitar de mais tempo para as atividades práticas com o robô, evidenciado pelas expressões “aula foi demasiado rápida” e “precisávamos de mais tempo para explorar as aplicações do robô”. Para além disso o comentário sugestivo “A única crítica é os braços do nosso robô, que apenas serviam para baralhar, visto que não funcionavam de maneira alguma”, pode ser indicativo que os alunos no final do Desafio 1 sentiram necessidade de obter maior controlo sob o robô, revelando querer fazer ações não suportadas pela programação direta no i-Brick, visto neste tipo de programação não ser possível controlar o terceiro motor

que fazia o robô mover os braços para cima e para baixo, conseguindo projetar bolas. O comentário “Não há sugestões a apresentar”, foi classificado como neutro e considerado como evidência que os alunos gostaram do desafio como foi proposto e assim não terem sugestões a apresentar.

Categorização das respostas obtidas na questão aberta do questionário sobre o Desafio 1

Comentário positivo

- “Foi giro.”
 - “Esperamos que seja tão interessante como está a ser até agora :)”
-

Comentário sugestivo

- “A única crítica é os braços do nosso robô, que apenas serviam para baralhar, visto que não funcionavam de maneira alguma.”
 - “A segunda aula foi demasiado rápida!”
 - “Achamos que precisávamos de mais tempo para explorar as aplicações do robô.”
-

Comentário neutro

- “Não há sugestões a apresentar.”

Anexo V: Questionário Desafio 2 e análise

Questionário de Equipas sobre o Desafio 2

*1. Número da Equipa:

- ☐ 1
☐ 2
☒ 3
☐ 4
☐ 5
☐ 6

*2. Opção da Equipa sobre o Desafio 2

	Não concordo totalmente	Não concordo parcialmente	Indiferente	Concordo parcialmente	Concordo totalmente
Gostei de programar o robô em NXT-G	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Achei difícil programar em NXT-G	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gostei mais de programar em NXT-G do que diretamente no brick	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gostei de participar na escrita da história para o Desafio 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gostei de criar pseudocódigo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Entendi a utilidade de criar pseudocódigo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Gostei do Desafio 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Achei difícil o Desafio 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Considero que aprendi conceitos de programação com o Desafio 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Considero que aprendi conceitos de algoritmos com o Desafio 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

*3. Comentário Geral / Sugestões

No final da quinta e última aula de prática de ensino supervisionada, foi aplicado um questionário às equipas de alunos sobre as suas perspetivas em relação ao Desafio 2, composto pelas três dimensões: preferências, dificuldades e aprendizagens. Em relação à dimensão de preferências das equipas em relação a programar o robô, escrever a história, criar pseudocódigo e experimentar o Desafio 1, a média mais elevada corresponde ao gosto por programar o robô em NXT-G com 3,83. Na verdade, os alunos revelaram ser indiferentes à escrita da história com média de 2,50 e gostaram pouco de criar pseudocódigo com média de 2,17. Salienta-se o facto de os alunos referirem que gostaram mais de programar em NXT-G do que diretamente no i-Brick, traduzido na média de 4,17 de preferência.

Dimensão preferências em relação ao Desafio 2

	Média	Desvio Padrão
Gostei de programar o robô em NXT-G	3,83	1,60
Gostei mais de programar em NXT-G do que diretamente no i-Brick	4,17	0,75
Gostei de participar na escrita da história para o Desafio 2	2,50	1,22
Gostei de criar pseudocódigo	2,17	1,33
Gostei do Desafio 2	3,67	1,51

Quanto aos níveis de dificuldades sentidos pelos alunos, este revelam não achar a programação em NXT-G nem o Desafio 2 em si muito difíceis com médias de 1,50 e 1,67. No entanto, salienta-se que estas médias são superiores em relação ao Desafio 1, o que era esperado devido ao nível de dificuldade, ou complexidades crescente imposto na planificação dos desafios.

Dimensão dificuldades em relação ao Desafio 2

	Média	Desvio Padrão
Achei difícil programar em NXT-G	1,50	0,55
Achei difícil o Desafio 2	1,67	0,52

Nota: Escala: 1 não concordo totalmente, 2 não concordo parcialmente, 3 indiferente, 4 concordo parcialmente e 5 concordo totalmente.

Apesar dos alunos considerarem este desafio um pouco mais difícil que o anterior, a nível de aprendizagens os alunos afirmaram ter conseguido aprender alguns conceitos de programação e algoritmos com o Desafio 2, expressos nas médias de 4.

Dimensão aprendizagens em relação ao Desafio 2

	Média	Desvio Padrão
Considero que aprendi conceitos de programação com o Desafio 2	4,00	0,63
Considero que aprendi conceitos de algoritmos com o Desafio 2	4,00	0,89

Em relação à questão aberta de comentário ou sugestão sobre o desafio, cinco foram categorizados como positivos, um como neutro e nenhum como sugestivo. Como comentários positivos, os alunos revelam “gostamos bastante” do Desafio 2, considerando-o “interessante”, “giro” e promotor de aprendizagens. O Comentário “Gumble na zona!”, foi classificado como positivo, porque *Gumble* foi o nome atribuído ao robô de uma das equipas, e ao afirmar que este estaria na zona, foi considerado como evidência de envolvimento. Como comentário neutro os alunos afirmam que não têm sugestões a fazer, podendo ser uma evidência de que gostaram do desafio assim como lhes foi proposto. A não existência de comentários sugestivos pode ser evidência que os problemas apontados no Desafio 1 relativamente à falta de tempo de experimentação do robô e maior controlo sobre o robô foram colmatados neste desafio. Mais especificamente em relação ao controlo do robô, nos comentários positivos os alunos afirmaram “Foi giro programar o robô e ver que ele faz o que exatamente programámos”.

Categorização das respostas obtidas na questão aberta do questionário sobre o Desafio 2

Comentário positivo

- “Não temos nada a acrescentar, gostamos bastante.”
- “No geral foi um desafio bastante interessante. Aprendemos alguns conceitos de programação aumentando assim o nosso conhecimento.”
- “Foi giro programar o robô e ver que ele faz o que exatamente programámos.”
- “Espero que o resto da experiência seja tão enriquecedora e divertida quanto este segundo desafio.”
- “Gumble na zona!”

Comentário neutro

- “Não há.”

Anexo W: Pós Questionário sobre atitudes e percepções e atenção

Questionário sobre atitudes e percepções e atenção (2)

A professora Diana Oliveira no âmbito da Prática de Ensino Supervisionada integrante do Mestrado de Ensino de Informática da Universidade de Lisboa, está a desenvolver um estudo sobre o impacto da robótica educativa no ensino e aprendizagem de programação à disciplina de Aplicações Informáticas B, mais especificamente aos conteúdos: conceitos de programação e algoritmos. Este questionário constitui-se parte fundamental do estudo e destina-se a recolher informação sobre as atitudes e percepções dos alunos em relação a aprender programação com robôs. O questionário leva aproximadamente 15 minutos a responder. O preenchimento é individual e as respostas de cada inquirido serão salvaguardadas por confidencialidade, servindo apenas para fundamentação empírica do estudo. Desde já agradeço a colaboração. Nota: entenda-se que no questionário a palavra “ansioso” é usada não como sinónimo de “estar nervoso”, mas como “desejar veemente”.

* 1. Primeiro e último nome

* 2. Expresse sua opinião sobre:

	Não concordo totalmente	Não concordo parcialmente	Indiferente	Concordo parcialmente	Concordo totalmente
Estou ansioso para aprender a programar.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Estou ansioso para aprender a programar com robôs.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Existe algo interessante que me chamou a atenção sobre aprender a programar com robôs.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A forma como o robô foi usado para ensinar programação ajudou a manter minha atenção.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Aprender a programar com o robô estimulou a	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

minha curiosidade.					
O trabalho que realizamos para aprender a programar com o robô prendeu minha atenção.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
A variedade de usos do robô ajudou a manter minha atenção.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Utilizar o robô para aprender a programar fez-me sentir recompensado por meu esforço.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Foi um prazer trabalhar com o robô para aprender a programar.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Eu gostei de usar o robô para aprender a programar tanto que eu gostaria de saber mais sobre ele.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

*** 3. Quais são os seus pensamentos e / ou sentimentos sobre aprender a programar?**

*** 4. Quais são os seus pensamentos e / ou sentimentos sobre o uso de robôs nesta disciplina?**

*** 5. Quais são os seus pensamentos e / ou sentimentos sobre como esta disciplina poderia ser melhorada?**

Anexo X: Grelhas de avaliação da prática de ensino supervisionada

Classificações finais do Desafio 1

		Individual		Equipa	Total	
		Programa	Interpretação	Reflexão		
Equipa 1	Aluno 1	95%	90%	90%	91%	18
	Aluno 2	80%	100%	90%	90%	18
	Aluno 3	80%	100%	90%	90%	18
	Aluno 4	95%	100%	90%	94%	19
Equipa 2	Aluno 5	85%	100%	93%	93%	19
	Aluno 6	90%	100%	93%	94%	19
	Aluno 7	95%	100%	93%	95%	19
	Aluno 8	85%	100%	93%	93%	19
Equipa 3	Aluno 9	95%	60%	57%	67%	13
	Aluno 10	100%	60%	57%	69%	14
	Aluno 11	100%	60%	57%	69%	14
	Aluno 12	80%	100%	57%	74%	15
Equipa 4	Aluno 13	95%	100%	48%	73%	15
	Aluno 14	90%	100%	48%	72%	14
	Aluno 15	95%	100%	48%	73%	15
Equipa 5	Aluno 16	80%	60%	62%	66%	13
	Aluno 17	95%	60%	62%	70%	14
	Aluno 18	80%	60%	62%	66%	13
Equipa 6	Aluno 19	75%	100%	63%	75%	15
	Aluno 20	75%	100%	63%	75%	15
	Aluno 21	75%	90%	63%	73%	15
	Aluno 22	75%	90%	63%	73%	15
	MÉDIA	87%	88%	70%	79%	

Classificações finais do Desafio 2

		Parte 1			Parte 2			Parte 3			Total	
		NXT-G	Video	Pseud	NXT-G	Video	Pseud	NXT-G	Video	Pseud		
Equipa 1	Aluno 1	100%	100%	75%	100%	0%	75%	100%	0%	75%	78%	16
	Aluno 2	100%	100%	75%	100%	0%	75%	100%	0%	75%	78%	16
	Aluno 3	100%	100%	75%	100%	0%	75%	100%	0%	75%	78%	16
	Aluno 4	100%	100%	75%	100%	0%	75%	100%	0%	75%	78%	16
Equipa 2	Aluno 5	100%	100%	100%	100%	100%	100%	0%	100%	0%	84%	17
	Aluno 6	100%	100%	100%	100%	100%	100%	0%	100%	0%	84%	17
	Aluno 7	100%	100%	100%	100%	100%	100%	0%	100%	0%	84%	17
	Aluno 8	100%	100%	100%	100%	100%	100%	0%	100%	0%	84%	17
Equipa 3	Aluno 9	100%	0%	0%	100%	100%	100%	100%	0%	75%	70%	14
	Aluno 10	100%	0%	0%	100%	100%	100%	100%	0%	75%	70%	14
	Aluno 11	100%	0%	0%	100%	100%	100%	100%	0%	75%	70%	14
	Aluno 12	100%	0%	0%	100%	100%	100%	100%	0%	75%	70%	14
Equipa 4	Aluno 13	100%	100%	80%	100%	100%	-	-	-	-	94%	19
	Aluno 14	100%	100%	80%	100%	100%	-	-	-	-	94%	19
	Aluno 15	100%	100%	80%	100%	100%	-	-	-	-	94%	19
Equipa 5	Aluno 16	100%	100%	95%	100%	100%	95%	100%	100%	80%	97%	19
	Aluno 17	100%	100%	95%	100%	100%	95%	100%	100%	80%	97%	19
	Aluno 18	100%	100%	95%	100%	100%	95%	100%	100%	80%	97%	19
Equipa 6	Aluno 19	100%	0%	80%	100%	100%	80%	100%	0%	80%	80%	16
	Aluno 20	100%	0%	80%	100%	100%	80%	100%	0%	80%	80%	16
	Aluno 21	100%	0%	80%	100%	100%	80%	100%	0%	80%	80%	16
	Aluno 22	100%	0%	80%	100%	100%	80%	100%	0%	80%	80%	16

Nota Final

		90%				10%	Finalíssimo	
		Desafio 1	Desafio 2	Extra	Total	Atitudes a valores		
Equipa 1	Aluno 1	18	16		17	17	17	Bom
	Aluno 2	18	16		17	16	17	Bom
	Aluno 3	18	16		17	16	17	Bom
	Aluno 4	19	16	0,5	18	18	18	Muito bom
Equipa 2	Aluno 5	19	17		18	18	18	Muito Bom
	Aluno 6	19	17		18	18	18	Muito Bom
	Aluno 7	19	17		18	19	18	Muito Bom
	Aluno 8	19	17		18	18	18	Muito Bom
Equipa 3	Aluno 9	13	14		14	17	14	Bom
	Aluno 10	14	14		14	17	14	Bom
	Aluno 11	14	14		14	16	14	Bom
	Aluno 12	15	14	-1	14	16	14	Bom
Equipa 4	Aluno 13	15	19	1	18	20	18	Muito Bom
	Aluno 14	14	19	1	18	20	18	Muito Bom
	Aluno 15	15	19	-2	15	15	15	Bom
Equipa 5	Aluno 16	13	19	1	17	17	17	Bom
	Aluno 17	14	19	1	18	17	17	Bom
	Aluno 18	13	19	-1	15	14	15	Bom
Equipa 6	Aluno 19	15	16		16	17	16	Bom
	Aluno 20	15	16		16	14	15	Bom
	Aluno 21	15	16		16	16	16	Bom
	Aluno 22	15	16		16	15	15	Bom

Visto grande parte da componente dos desafio ter sido desenvolvida em equipa, de modo a obter uma avaliação o mais representativa do trabalho desenvolvido por cada aluno, em dialogo em a professora cooperante, foi acrescentado às classificações finais um fator de correção de no máximo dois valores, com as seguintes explicações que foram dadas a conhecer aos alunos:

- Equipas 4 e 5 têm um valor extra por terem ido mais além no desafio 2. A equipa 4 tinha um exercício mais difícil do que as outras equipas e conseguiu juntar os 3 programas num só. A equipa 5 juntou os três programas em 2 e para além disso explorou blocos como o display e sound de forma autónoma.
- O aluno S tem meio valor extra porque no desafio 1 foi o aluno com a melhor interpretação do comportamento do robot, e usou a sua forma de interpretação no programa de equipa.
- Os valores negativos no critério extra, correspondem a alunos que nas componentes de equipa foram menos participes. Facto comprovado por perguntas ao longo das aulas e pela sua pouca participação/iniciativa.

Anexo Y: Análise das questões abertas do pós questionário

Análise da questão aberta:

Quais são os seus pensamentos e / ou sentimentos sobre aprender a programar?

N.º Comentários positivos

- 1 Aprecio bastante, programar é divertido, se for feito com um objetivo, eu espero que continuemos a programar como temos vindo a fazer, visto que é eficaz.
- 2 Acho que é interessante programar e no futuro acho que é útil em várias áreas de conhecimento.
- 3 Imensa curiosidade a possibilidade de aumentar a complexidade da programação e dos resultados que possam daí surgir.
- 4 Pensamentos bons.
- 5 Querer melhorar.
- 6 Gostei bastante.
- 7 Gostei.
- 8 Estou ansioso, vai ser fixe! :D
- 9 Participando a minha opinião, tenho de admitir que achei bastante entusiástico, todo um processo de aprendizagem que nos impingiu, no bom sentido é claro, de uma forma diferente e interessante, os básicos da programação, para mais tarde, envergarmos para situações de uma complexidade maior. Sinto-me por isso bastante agradado com o decorrido até então!
- 10 Fixe.
- 11 Ansio o dia em que possa programar bem.
- 12 Positivo, ansioso.
- 13 Positivo.

N.º Comentários negativos

- 1 Seria, de facto, algo de bastante interesse, mas não sabendo até que ponto tal assunto poderia ser aprofundado, a minha curiosidade é-lhe diretamente proporcional.
- 2 Parece interessante mas prefiro a parte prática que a teórica.
- 3 Acho que era interessante, especialmente se aprendesse-mos uma linguagem útil como: C, C#, C++, LUA, Javascript, etc.. mas era giro mesmo que aprendêssemos BASIC ou algo do género!

Análise da questão aberta:

Quais são os seus pensamentos e / ou sentimentos sobre o uso de robôs nesta disciplina?

N.º Comentários positivos

- 1 Torna as coisas mais dinâmicas, se estivesse somente a copiar dados do quadro, ou escrever as explicações dos professores as aulas seriam aborrecidas. Trabalhar assim com os robôs é bastante melhor, e permite-nos ganhar experiência e conhecimentos teóricos, em vez de somente a teoria.
- 2 Acho que estimula os alunos a ficarem mais interessados na disciplina.
- 3 O aumento do interesse na programação.
- 4 Boas *vibes*.
- 5 Torna a disciplina mais interessante.
- 6 Ajuda a manter o interesse e a querer aprender sempre mais.
- 7 Acho interessante visto que segundo o programa iríamos aprender uma básica "iniciação à programação". Nada mais empolgante que robôs!
- 8 Muito interessante.
- 9 Gostei bastante, estimulou o meu gosto por programação.
- 10 Para colmatar, tudo o que foi expresso na resposta anterior, os robôs acentuaram, mais uma vez, a criatividade deste programa de cinco aulas, do qual não me arrependo nada de ter tido. Assim, os robôs foram, de certa forma, a forma mais divertida que me ocorre de testar o aprendido na bela linguagem que é a programação. Sinto-me, então, também bastante agradado
- 11 É um bom método de introdução à programação.
- 12 Não me importo de programar robôs, até é giro.
- 13 Penso que é uma maneira bastante intuitiva de ensinar os conceitos mais básicos de programação. Os alunos podem assim aspirar a lógicas mais avançadas, quiçá, serem programadores no futuro.
- 14 Os robôs ajudam a prender a atenção e a definir conceitos importantes sobre programação.
- 15 Positivo.

N.º Comentários negativos

- 1 Não totalmente o que pensava (pensei mais na parte do programa no pc)

Análise da questão aberta:

Quais são os seus pensamentos e / ou sentimentos sobre como esta disciplina poderia ser melhorada?

N.º Comentários positivos

- 1 Com os meios possíveis neste momento acho que não pode ser melhorada.
- 2 Acho que no geral está a ser bem lecionada.
- 3 *Bacano.*
- 4 Está bem como está.
- 5 Positivo.

N.º Comentários negativos

- 1 Provavelmente, seria melhor se tivéssemos mais tempo de aula ou mais espaço para realizar as experiências. Fora disso não me ocorre nada.
- 2 Através de atividades práticas, como mexer no robô.
- 3 Não usar o basquetebolista!!!!
- 4 As minhas únicas críticas são em relação a alguns problemas com o robô que o meu grupo experimentou, de resto estou bastante contente com o rumo que a disciplina tomou.
- 5 O aluno x é não pára quieto.
- 6 O aluno y é um elemento destabilizador da turma e da equipa.
- 7 Penso que, à parte de tudo o resto que já escrevi, poderíamos, eventualmente, ter aprofundado um pouco mais os conteúdos da programação. Posto isto, nada mais tenho a apontar.
- 8 Menos perguntas.
- 9 Desconhecendo o resto do programa da disciplina, não sei até que ponto estou a divagar, mas penso que faria sentido que os assuntos já tratados, e outros, fossem mais aprofundados.
- 10 Aprecio mais a parte prática e, apesar de considerar importante os power points como modo de desenvolvimento de conceitos, preferia que não se tivesse de preencher tantos power points e criar tantas histórias.
- 11 Aprender uma linguagem de programação útil.
- 12 Mais detalhado no programa de pc.

Anexo Z: Complementos da narrativa da prática de ensino supervisionada

Reflexão Inicial sobre Conceitos de programação – primeira aula

Dei início à Reflexão Inicial sobre Conceitos de Programação, com a finalidade de recolher informações diagnósticas sobre o que os alunos sabiam sobre programação e completar esse conhecimento. Na caracterização da turma tinha recolhido a informação que alguns alunos já haviam programado, mas queria saber que conceitos de programação haviam aprendido dessas experiências. Com o auxílio do segundo slide do Anexo L, referi que nas informações recolhidas da turma havia reparado que muitos afirmaram querer seguir a área de informática. Perguntei se algum deles queria ser programador, ao que nenhum reagiu. Então perguntei se sabiam o que fazia um programador no seu quotidiano. Apenas um aluno respondeu: “Programa”. Neste momento senti que teria que quebrar gelo e estimular a participação da turma, e que seria melhor fazer perguntas mais fechadas e específicas. Logo, decidi fazer uma pergunta extra a cada equipa, sobre se achavam a profissão de programador difícil. Considerei positivo ouvi-los expressar-se sobre os seus pensamentos e ansiedades, e finalmente vê-los interagir comigo. No entanto, todas as equipas revelaram ter um conceito negativo da programação considerando-a difícil, trabalhosa, envolvendo muitas horas de trabalho. Apenas uma equipa referiu que a dificuldade de uma tarefa dependia muito se a pessoa estava a fazer algo que gostava, o que me reportou para a importância de criar contextos de aprendizagem apelativos para que os alunos sentissem vontade de superar as suas dificuldades. Face a estas respostas, nas minhas notas pessoais escrevi que deveria certificar-me do grau de dificuldade dos meus desafios de modo a que os alunos não subentendessem que a informática fosse difícil devido a uma possível excessiva exigência minha e não pela informática em si. Recorrendo ao slide 3 do Anexo L fiz perguntas específicas que conduziram os alunos a chegar ao conceito do que era um programa e uma linguagem de programação, para que servia a programação e com que objetivos iriam programar. Reforcei positivamente os alunos por dizer que programar era mais simples do pensavam. Durante esta reflexão inicial apercebi-me que alguns elementos das equipas 5 e 6 distraíam-se mutuamente, arrazoei com eles que era importante para eles e para a turma que fizessem silêncio e prestassem atenção. Para os ajudar nesta tarefa, tentei envolvê-los na reflexão, solicitando a participação dos elementos mais perturbadores na exploração de alguns conceitos. Assim consegui ter a turma mais atenta.

Auto e h tero avalia  o - terceira aula

A equipa 6 revelou-se consciente de si mesma por expressar como ponto menos forte o comportamento. Foi assumido um compromisso de melhoramento do comportamento, sendo que este afetava o seu desempenho. Seguidamente foi elogiado o bom trabalho desenvolvido pela equipa 1. Salientando que a n vel de programa  o o Aluno S havia sido o foi mais forte, facto que os colegas de equipa n o reconheceram. Como aspeto menos positivo, deveriam participar mais nas reflex es de turma. A auto avalia  o da equipa 5 coincidiu com a minha. Quanto   equipa 2, esta n o concordava que o seu aspeto menos forte seria a participa  o. Afirmei que participar envolvia n o s o responder  s minhas quest es, mas responder com assertivamente e voluntariedade. Assumi compromisso que iria estar mais atenta  s suas interven  es e salientei “  bom voc es saberem qual   a perce  o que eu tenho sobre voc es. Porque, reparem voc es acham que s o participativos e eu n o notei muito isso. Assim, eu vou estar com mais aten  o e voc es t m v o-se esfor ar um bocadinho mais por participarem. Temos um acordo.” O aluno G, que na aula dois havia assumido que tinha receio em chamar-me quando tinha d vidas, foi o porta-voz do grupo e afirmou “O ponto fraco talvez seja tirar d vidas. Mas isso   bom. Eu n o acho que isso seja fraco, acho que se isso mostra que...” devido aos risos dos colegas n o continuou. Eu reforcei usando como exemplo a aula anterior que se o aluno me tivesse chamado mais cedo n o teria perdido tanto tempo na resolu  o do exerc cio. Ao que o aluno concordou dizendo que sim, mas por outro lado expressou-se referindo “Mas   o orgulho”. Reconforte o aluno dizendo que se eles soubessem tudo n o precisariam de mim. E que o conhecimento era algo que se constru a aos poucos, e que o n o saber e descobrir faz parte do processo. O aluno concordou mais uma vez. Ao dirigir-me para a equipa 4 o aluno M abordou-me com a quest o “Mas   bom ter d vidas professora?”. Ao que afirmei que era bom ter d vidas, e que mau seria ficar com elas. O aluno continuou “Mas e se n s tirarmos as d vidas n s pr prios?” Referi que era bom, mas que a quest o neste momento era se ele o conseguiria sozinho. Respondendo prontamente que sim. Alerttei-os para o facto de quando fazemos algo que pensamos que est  certo, n o quer dizer que o esteja e que o seu excesso de confian a na  ltima aula, em n o querer rever o trabalho fez com que tivessem m  nota nessa componente. Corrigi brevemente com eles essa componente. Os alunos perceberam onde erraram e afirmaram que deveriam ter aproveitado a minha disponibilidade. Feita a auto e h tero avalia  o, dei in cio   Reflex o inicial sobre conceitos de algoritmos. Pedi a intera  o dos alunos e

questionei as equipas 1 e 2 se queriam melhorar a sua componente de participação. Ao que um elemento da equipa 1 disse que não. Retorqui que pelos vistos eu teria razão ao dizer que eles não eram participativos. Os restantes elementos da equipa discordaram aquele aluno, e afirmaram que queriam responder. Afirmei que tinham perdido aquela oportunidade que iria dá-la à equipa 1, mas que iriam haver mais e que se eles queriam ter mais cotação na participação tinham que me mostrar. A partir deste momento a equipa 1 passou a ser uma das equipas mais interventivas nas reflexões de turma. Visto a auto e hétero avaliação ter demorado mais do que o previsto sobrou menos tempo para a atividade extra, realizada durante a reflexão inicial. Notei que não era fácil para os alunos refletirem sobre as características dos algoritmos, sendo necessário apoiar e conduzir as equipas nesta reflexão.

Reflexão Inicial sobre conceitos de programação e algoritmos - quarta aula criação do contexto de aprendizagem

Disse à turma que nesta aula em conjunto teriam a oportunidade de criar o enredo a história que iriam experienciar. E com base do slide 8 do Anexo P, li a parte inicial do enredo “No horizonte distante apercebem-se os contornos de um imponente castelo. Um desejo crescente de vencer este desafio percorre o coração dos heróis, no entanto a dificuldade do trajeto, fá-los subitamente duvidar de si mesmos. Na mente de todos pairava intermitentemente a questão: Será que vamos conseguir?”. Seguidamente, pedi aos alunos de verificassem a característica do seu robô, correspondente ao papel verde do envelope e perguntei “Quem tem uma característica indicada para responder à pergunta se vão conseguir e animar a história? A equipa 1 revela à turma ter a característica positivo. Então pedi-lhes que dessem uma mensagem positiva aos colegas, ao que responderam “Vamos a isso equipas! Vamos conseguir!”. Perguntei de seguida “Quem inicia o desafio?” A equipa 5 salienta que como tendo a característica corajoso, o seu robô teria a coragem de iniciar o desafio e “enfrentar o terrível Lama”. Questionados sobre quem deveria fazer o percurso com o robô corajoso, ninguém se manifestou. Então em conjunto fomos construindo a história contando-a oralmente, até ao ponto que era necessário decidir como iriam avançar a muralha, visto a terrível Lama encontrar-se precisamente à porta da muralha. A equipa 5 afirma que sendo a seu robô corajoso iriam tentar empurrar a Lama derrubando-o, a turma relembra-os que o Lama tem a capacidade de lançar bolas e que assim o seu robô iria ferir-se, ao que a equipa 2 reage e afirma ter a característica

médico referindo que se os braços do robô mexessem podiam socorrer o robô corajoso. Disse aos alunos então, que continuava por resolver o dilema de como iriam ultrapassar a muralha. Face este dilema foi perguntado à turma que equipa teria a característica ideal para soluçona-lo, ao que a equipa 6 revelou possuir a característica génio. Visto não poderem atravessar a muralha os alunos apresentaram como solução “Derrubamos a muralha”. Perguntei-lhes como o fariam e sugeri que analisassem bem a muralha.



Muralha do Desafio 2

A equipa 6 ao analisar a muralha com mais atenção repara que na extremidade direita ela é constituída por caixas mais pequenas, referindo que estas poderiam ser o sinónimo de que a muralha estava a cair e assim estar fragilizada naquele ponto, sendo o ponto ideal para a derrubar. A turma aceitou e gostou da solução proposta, então afirmei “O Génio já descobriu a solução, mas quem é que a vai concretizar?”. “O corajoso!” afirma de novo a equipa 5 e eu referi que a característica deles já havia sido usada, assim teria que ser outra equipa. Depois de algum tempo de pausa a equipa 3 revela ter a característica forte, que era indicada para conseguir derrubar a muralha. Perguntei por fim que característica faltava ao que a equipa 4 diz ser a sua de assustadiço e que fazia sentido o robô deles na história ser o último a percorrer o cenário. Então sugeri que o assustadiço com medo ao atravessar a muralha chamaria a atenção do Lama. Ao que a equipa sugere o seu robô ao ver o Lama assusta-se e emite um som que faz o Lama perceber que tinham avançado a muralha. Então afirmo que cada um dos robôs tinha sido escolhido por ter uma habilidade especial e sugeri que no final a usassem para imobilizar ou afugentar a Lama. Os alunos sorridentes questionam-me se a cauda do Spike e os braços do basquetebolista e Tribot mexiam, ao que os desafio para experimentarem.